



# Caractérisation et Reconstruction de Solides Tridimensionnels par Squelette Ellipsoïdal

Frédéric Banegas

## ► To cite this version:

Frédéric Banegas. Caractérisation et Reconstruction de Solides Tridimensionnels par Squelette Ellipsoïdal. Géométrie algorithmique [cs.CG]. Ecole Nationale Supérieure des Mines de Saint-Etienne; Université Jean Monnet - Saint-Etienne, 2000. Français. NNT : 2000STET4020 . tel-00985909

**HAL Id: tel-00985909**

**<https://theses.hal.science/tel-00985909>**

Submitted on 30 Apr 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Jean Monnet de Saint-Etienne (UJM)  
Ecole Nationale Supérieure des Mines de Saint-Etienne (ENSMSE)  
Centre International de Recherche en Agronomie pour le Développement (CIRAD)  
Centre Hospitalier Universitaire Lapeyronie de Montpellier (CHU Lapeyronie)

Année 2000

**THESE**

Pour obtenir le grade de  
Docteur en Informatique

# **Caractérisation et Reconstruction de Solides Tridimensionnels par Squelette Ellipsoïdal.**

Préparée conjointement au CIRAD et à l'ENSMSE

Présentée et soutenue publiquement par

**Frédéric Banégas**

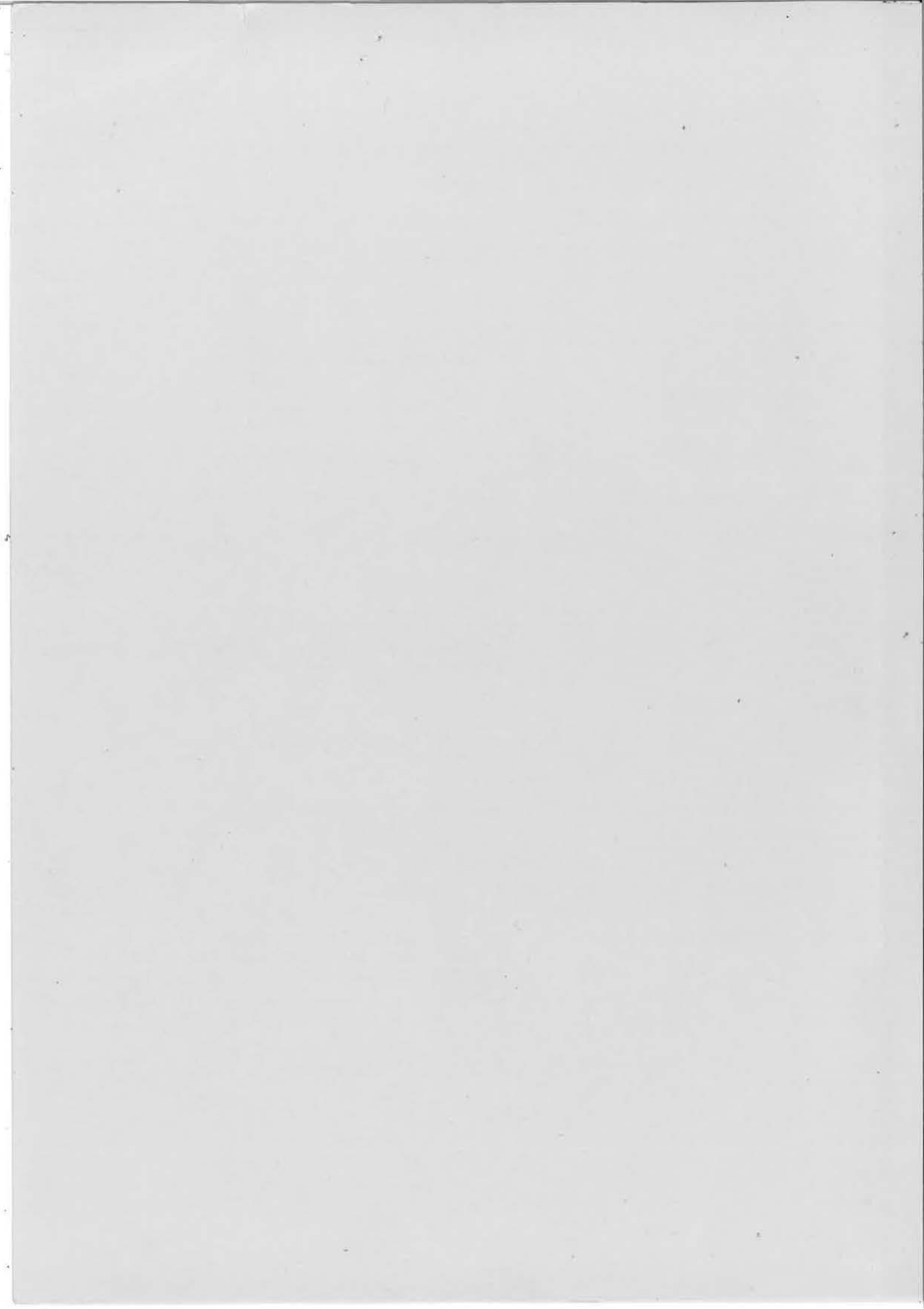
Le 9/11/00



**Composition du Jury**

Marie-Paule Cani Rapporteur  
Philippe Cinquin Rapporteur  
Colin de la Higuera Examineur  
Marc Roelens Examineur  
Marc Jaeger Examineur  
Dominique Michelucci Examineur  
François Canovas Membre Invité









Université Jean Monnet de Saint-Etienne (UJM)  
Ecole Nationale Supérieure des Mines de Saint-Etienne (ENSMSE)  
Centre International de Recherche en Agronomie pour le Développement (CIRAD)  
Centre Hospitalier Universitaire Lapeyronie de Montpellier (CHU Lapeyronie)

**Année 2000**

3 4200 00772927 6

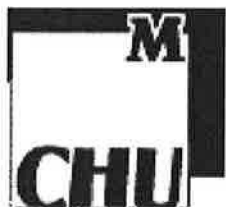
**THESE**

Pour obtenir le grade de  
Docteur en Informatique

# **Caractérisation et Reconstruction de Solides Tridimensionnels par Squelette Ellipsoïdal.**

Préparée conjointement au CIRAD et à l'ENSMSE

Présentée et soutenue publiquement par  
**Frédéric Banégas**  
Le



28 JAN. 2002

PPN 060385448

EPRN 149156871

EXL 334 837

**Composition du Jury**

Marie-Paule Cani Rapporteur  
Philippe Cinquin Rapporteur  
Colin de la Higuera Examineur  
Marc Roelens Examineur  
Marc Jaeger Examineur  
Dominique Michelucci Examineur  
François Canovas Membre Invité



# **Résumé / *Abstract.***



## Résumé

Le volume des données décrivant les solides tridimensionnels est sans cesse en augmentation. Une des principales causes de ce phénomène est le gain en résolution des modalités d'acquisition, qu'elles soient volumiques ou surfaciques. Par voie de conséquence, l'analyse de ces informations devient de plus en plus coûteuse et lourde à mettre en œuvre, requérant des capacités de traitement ou de stockage importantes.

Une phase de traitement, en aval du processus de numérisation est indispensable : les données doivent être structurées non seulement pour accélérer leur gestion mais aussi pour améliorer et favoriser la compréhension de leur contenu. Enfin, cette phase se doit d'être flexible afin d'être applicable à des domaines variés d'expertise.

Nous proposons au travers de ce travail de thèse un modèle permettant de transformer tout solide numérisé en une entité répondant aux critères énoncés précédemment. Le *Squelette Ellipsoïdal* (ou E-Squelette) décompose les structures géométriques en sous-structures pertinentes tout en prélevant les informations importantes aux yeux de l'expert. En sortie de ce processus, on dispose de capacités de vision multi-échelle de la géométrie, enrichie des informations extraites. Des comparaisons dépendant de l'échelle de vision sont autorisées entre E-Squelettes, assurant le suivi temporel et la reconnaissance automatique d'objets solides 3D. Enfin, les données présentent de très importants taux de compression, et peuvent être transmises de manière progressive par réseau. L'erreur d'approximation est mesurable et contrôlable en terme de géométrie, ce qui garantit le contrôle de la validité des données transformées.

Nous montrerons comment le E-Squelette s'applique à l'imagerie médicale et peut enrichir les moyens d'observation et de diagnostic.

## Abstract

The amount of data needed to describe both volume and surface of 3D objects is often huge and produces bottlenecks at every step of analysis. Thus, extracting relevant information in this case demands heavy and complex processing techniques. A preprocessing phase is dramatically required : data must be synthesized in order to enhance analysis procedure by providing context-dependent measurements. Parameters set should not be rigid, as it can be radically different from one application to another.

The method we propose in this PhD. thesis consists in mapping any digitized 3D solid – taking into account its inner points – into *Ellipsoidal Skeleton* (or E-skeleton). Based on binary shape decomposition into a union of simple sub-shapes paradigm, it also gathers relevant information about the geometry and any other set of values that seems interesting, depending on the study context. Each sub-shape and its parameters set are more generically viewed as a *feature*, which is assumed to be non-decomposable at a given *semantic level*. This semantic zoom capability for object description permits a hierarchical approach, i.e. a scale of vision control. Low semantic zoom allows crude approximation for fast pre-classification while high semantic zoom highlights finer details for precise comparison.

From this preprocessing stage, tasks such as object recognition and object analysis are made easy and intuitive to perform via features comparison. Any bottleneck is removed, ensuring fast data processing. At last, the versatile structure of the E-skeleton allows not only future improvement but also practical ways to add features to the E-skeleton repertoire.

We will show how the E-skeleton is well-suited for medical imaging and how it can be seamlessly integrated into the analysis and diagnostic process.





# **Table des Matières.**



## 0. Introduction.

1. Motivations.....	14
2. Cadre de travail.....	15
3. Plan du rapport.....	16

## 1. La Réalité Virtuelle.

1. Introduction.....	19
1.1. Qu'est-ce que la Réalité Virtuelle ? .....	19
1.2. Un peu d'histoire... ..	20
2. L'Immersion, qu'est-ce que c'est ?.....	26
2.1. Les informations sensorielles.....	26
2.2. Les informations proprioceptives .....	27
2.3. La Combinaison des Informations est la clé du succès.....	29
3. Les outils de l'immersion.....	33
3.1. Voir et Entendre.....	33
3.2. Toucher.....	36
3.3. Se Déplacer.....	38
4. Et le mal de mer dans tout ça ? .....	41
4.1. Ce que l'on a observé... ..	41
4.2. Quels sont les symptômes ? .....	42
4.3. Quelles sont les causes ? .....	42
4.3.1. L'Individu.....	43
4.3.2. Le Simulateur en lui-même (le matériel mis en œuvre).....	44
4.3.3. La tâche à réaliser dans le monde virtuel.....	45
4.3.4. Conclusion .....	46
5. La RV et le monde médical.....	47
5.1. Les apports de la RV.....	47
5.2. Les impératifs de l'imagerie médicale.....	48
5.3. Les Appareillages.....	48
5.4. Les Applications.....	49
6. Conclusion(s).....	52
6.1. Le futur de la RV, pour elle-même et pour la médecine.....	52
6.2. Quel impact sur la société ?.....	55
6.3. La RV, une nouvelle réalité oui mais... le « <i>willing suspension of disbelief</i> ».....	56
7. Glossaire.....	59
8. Quelques adresses internet pour aller plus loin.....	66
9. Bibliographie.....	67

## 2. Tour d'Horizon de la Représentation Tridimensionnelle.

1. Introduction.....	70
2. Représentation volumique.....	71
2.1. Acquisition volumique des données.....	71
2.2. Modèles internes de représentation volumique.....	72



2.2.1. Les voxels.....	72
2.2.2. Les surfaces implicites.....	72
2.3. Projection volumique des données. ....	76
3. Représentation surfacique des objets. ....	80
3.1. Acquisitions surfaciques.....	80
3.2. Modèles internes surfaciques.....	81
3.2.1. La représentation polygonale directe. ....	81
3.2.2. Les fonctions explicites. ....	82
3.2.3. La représentation paramétrique. ....	83
3.3. L'allègement des données par simplification surfacique .....	89
4. Conclusion.....	93
5. Bibliographie.....	94

### 3. Tour d'Horizon de la Reconstruction Surfacique.

1. Introduction.....	98
2. La reconstruction surfacique.....	100
2.1. Reconstruction surfacique directe.....	101
2.1.1. Représentation explicite du contour : construction directe. ....	102
2.1.2. Représentation analytique du contour : construction indirecte. ....	107
2.2. Reconstruction surfacique par optimisation numérique ou modèle physique. ...	109
2.2.1. Obtention de la solution initiale.....	110
2.2.2. Les techniques d'optimisation numérique classiques. ....	110
2.2.3. Les Métaheuristiques.....	115
3. Conclusion.....	118
4. Bibliographie.....	119

### 4. Tour d'Horizon de la Reconnaissance de Formes.

1. Introduction.....	123
2. Approches pour la reconnaissance de formes. ....	125
2.1. Reconnaissance par motif. ....	125
2.2. Reconnaissance par parties pertinentes. ....	127
2.3. Reconnaissance structurelle.....	128
2.4. La reconnaissance statistique.....	130
3. La prise de décision ou comment mesurer la correspondance.....	131
3.1. Qu'est-ce qu'un attribut ?.....	131
3.2. La mesure de distance.....	131
3.3. La logique floue.....	134
3.4. Les réseaux neuronaux (reconnaissance statistique). ....	136
3.4.1. Qu'est-ce que c'est ? .....	136
3.4.2. Un exemple simple. ....	137
3.4.3. Perfectionnements et déboires des réseaux de neurones. ....	138
3.4.4. Quand utiliser les RNAs ? Avantages et inconvénients. ....	139
4. Conclusion.....	141
5. Historique de l'intelligence artificielle de 1637 à 1992 : mythes et réalités... ..	142



6. Bibliographie.....	146
-----------------------	-----

## 5. Le E-Squelette.

1. Introduction.....	150
2. Motivations.....	151
3. Formalisme et Outils Théoriques du E-Squelette. ....	152
3.1. Définition formelle du E-Squelette.....	152
3.1.1. Classes et sous-classes composant le E-Squelette.....	152
3.1.2. Construction des classes du E-Squelette : les nuées dynamiques.....	154
4. Construction du E-Squelette. ....	159
4.1. Considérations à propos de l'espace à partitionner. ....	159
4.2. La fonction de représentation. ....	159
4.2.1. Définition de la matrice de dispersion.....	159
4.2.2. Choix du représentant de classe.....	166
4.3. La fonction de distance.....	167
4.3.1. Les métriques. ....	167
4.3.2. Choix de la fonction de distance D . ....	169
4.4. Le critère d'optimisation W.....	169
4.5. Minimisation du critère d'optimisation W.....	170
4.6. L'Algorithme <i>progressif</i> de construction du E-Squelette. ....	171
4.6.1. La solution initiale, point de départ des nuées dynamiques.....	171
4.6.2. Découpage d'une classe en deux sous-classes. ....	172
4.6.3. Le critère de scission.....	173
4.7. Propriétés de l'algorithme progressif.....	176
4.8. Complexité de l'algorithme progressif.....	176
5. Le Modèle de Données du E-Squelette.....	178
6. Résultats. ....	180
6.1. Mesures de stabilité.....	180
6.1.1. Présentation des jeux de données. ....	180
6.1.2. Présentation des indicateurs de variation.....	182
6.1.3. Résistance à la perturbation en position/orientation.....	182
6.1.4. Résistance au Sous-échantillonnage. ....	183
6.1.5. Résistance aux erreurs de numérisation/segmentation.....	184
6.2. Evolution de la variance intraclasse totale : un critère d'optimalité ?.....	188
6.3. Temps de Calcul.....	189
7. Conclusions. ....	190
8. Bibliographie.....	191

## 6. Reconstruction Géométrique par E-Squelette.

1. Introduction.....	193
2. Modèle interne géométrique.....	194
2.1. Les primitives.....	194
2.2. Les surfaces implicites.....	196
2.3. L'Arbre Géométrique du E-Squelette. ....	200





2.4.	L'Ajustement Surfaccique.....	203
2.4.1.	Obtention de la Solution Initiale.....	203
2.4.2.	Choix de la primitive.....	205
2.4.3.	Un modèle « local » de déformation.....	205
2.4.4.	Fusion des primitives.....	206
2.4.5.	Evaluation de l'approximation géométrique.....	206
2.4.6.	Une méthode d'ajustement surfaccique : la méthode tabou.....	207
2.4.7.	Temps de Calcul.....	210
2.4.8.	Récapitulatif de l'algorithme de reconstruction.....	212
3.	Représentation géométrique explicite.....	214
3.1.	La polygonisation ou discrétisation.....	214
3.2.	Zoom sémantique et niveaux de détails.....	216
3.2.1.	Le zoom sémantique (ZS).....	216
3.2.2.	Les niveaux de détails.....	217
3.2.3.	Combinaison Dynamique.....	218
3.3.	Compression de Données et Transmission Progressive par Réseau.....	221
4.	Conclusion.....	224
5.	Bibliographie.....	225
<b>7.</b>	<b>Reconnaissance de Formes par E-Squelette.</b>	
1.	Introduction.....	229
2.	Rappel de la définition formelle du E-Squelette.....	230
3.	Le pipeline de mesure.....	231
3.1.	Pourquoi un pipeline ?.....	231
3.2.	Extraction de paramètres caractéristiques : les fonctions d'extraction.....	232
3.3.	Fonctionnement du pipeline de mesure.....	232
3.3.1.	La fonction de normalisation $f_N$ .....	232
3.3.2.	L'Appariement optimal.....	233
3.3.3.	Distance entre deux classes.....	234
3.3.4.	La matrice de distance $M_d$ .....	234
3.4.	Implantation de l'appariement optimal.....	234
3.4.1.	Initialisation de l'algorithme.....	235
3.4.2.	L'algorithme de maximisation de flux pour un coût minimal.....	236
3.5.	Récapitulatif.....	237
4.	Résultats.....	238
4.1.	Comparaison d'objets.....	238
4.1.1.	La comparaison progressive.....	238
4.1.2.	Le suivi.....	242
4.2.	Reconnaissance d'objets.....	246
5.	Conclusion.....	250
6.	Bibliographie.....	251
<b>∞.</b>	<b>Conclusion.</b>	

Annexe 1. Publications.

Annexe 2. Bibliographie Globale.



# **Avant-propos.**



Ce mémoire ne s'adresse en aucune façon à un public exclusivement familier de l'informatique. Même si certaines parties restent très techniques, la substance est à la portée d'un public large : elle peut être appréhendée par toute personne curieuse des thèmes traités ici.

Le chapitre 1 est absolument ouvert à tous puisqu'il traite de la réalité virtuelle sous un jour didactique et initiatique. Il constitue une certaine digression par rapport au thème de cette thèse, mais sa présence est un choix personnel qui a le mérite de montrer les limites d'une science que l'on croit mûre, notamment en ce qui concerne son adéquation avec le monde médical. Les parties également accessibles sont constituées des chapitres 2, 3 et 4 qui présentent le travail, les possibilités et les limitations actuelles de la visualisation, la reconstruction surfacique et la reconnaissance automatique de formes. Quant aux autres parties (5, 6 et 7), elles exposent en détail le modèle de E-Squelette : démonstrations, expérimentations et algorithmiques y sont omniprésentes.

Nous avons choisi de fournir un grand nombre d'illustrations et de légendes explicites. De cette manière, un survol rapide permet d'appréhender facilement les points traités.

Les références comportent un très petit nombre de publications qui ne sont pas citées dans le texte. Ces publications sont éloignées du sujet mais ont néanmoins concouru à parfaire certains aspects de notre recherche. Pour le lecteur décidé, elles pourront représenter une bonne entrée en matière, comme les ouvrages traitant de la psychologie cognitive par exemple.

Enfin, les algorithmes et techniques sont décrits le plus précisément possible afin que le mémoire seul puisse suffire à les reconstituer.





**Pensées.**

*A papy Jean,*





Peut-être ai-je tort mais je pense fermement que notre personnalité résulte entre autre de la somme d'une partie de celles des gens que nous connaissons, à laquelle s'ajoute un terme (presque) négligeable qui est notre moi profond. Bien sûr, si l'on n'y prend garde, cette partie de l'addition, considérée comme faible au départ, peut prendre des proportions étonnantes, chacun apportant son moi dans l'opération. Je vous ferai grâce d'un modèle de simulation numérique pour vérifier le phénomène, tant il serait biaisé au vu de l'imprécision numérique inhérente aux ordinateurs qui nous entourent (nous gouvernent ?) Est-ce ainsi qu'ils supprimeront notre personnalité ?

En tout cas, cette somme de « mois » fut grande pour ma part. J'ai dans ce travail reçu l'assistance d'un grand nombre de personnes : familles, amis et collègues, ce partitionnement étant le plus souvent flou, notamment pour les deux dernières classes – je n'ai en effet adopté personne (ni me suis fait adopter d'ailleurs) pendant ma thèse. Je veux ici remercier chaque personne qui m'a accompagné, car elles le méritent même si elles jureront le contraire.

Mes premiers remerciements s'adressent tout d'abord à messieurs Philippe de Reffye et Bernard Péroche. Ce sont eux qui ont instigué le projet et mis en place la convention entre le CIRAD et l'ENSMSE, me donnant ainsi les moyens de finaliser mon cursus par cette thèse de doctorat, et mettant à disposition les ressources communes des deux établissements.

Je remercie immensément Dominique Michelucci et Marc Roelens de l'ENSMSE qui ont encadré mon travail de doctorat, toujours disponibles malgré la distance Montpellier-Saint-Etienne qui nous séparait. J'admire leur technique d'aiguillage et de conseils, ils m'ont souvent donné l'impression que c'était moi qui avait eu l'idée. A cette équipe j'ajoute Marc Jaeger qui m'encadrerait officiellement au CIRAD, et plus officieusement – mais tout à fait entièrement pour ma part – Philippe Borianne. Ils sont la clé de tout le travail présenté ici et ont apporté une aide inestimable dans mes recherches. Le rôle de représentants de commerce ne leur a pas fait peur puisqu'ils ont répandu ces travaux à travers le monde, jusqu'en Chine. Merci à François Houllier qui a sans cesse veillé à ce que tout se passe pour le mieux et dans les meilleures conditions possibles au CIRAD. Je remercie le docteur François Canovas, chirurgien orthopédique à l'hôpital Lapeyronie de Montpellier, qui m'a fait l'honneur de travailler avec moi et à qui j'ai dû sans conteste enlever quelques illusions sur la puissance de l'outil informatique.

J'ai également mis à contribution mes amis, qui ont donné avec enthousiasme leur temps – et leur patience – pour combler mes lacunes en mathématiques. Je veux ici remercier Christophe Combescure, devenu doctorant cette année, qui m'a donné des cours en statistiques et probabilités, et Bianco Florian, chantre de la topologie et des casse-tête chinois.

Je réserve une place toute particulière à Stéphane Chemouny qui a réalisé sa thèse en même temps que moi et dans le même bureau. Il a instillé beaucoup d'idées dans mon



esprit, tant pour mon travail que pour le reste. Je pense lui devoir bien plus qu'il ne le pense (à moins qu'il ne le sache déjà). Merci à toi qui est mon ami aujourd'hui.

Un grand merci à toute l'équipe du laboratoire LISSE de l'ENSMSE, toujours présente autour de moi lors de mes visites à Saint-Etienne. L'équipe du CIRAD AMIS, avec laquelle j'ai passé la majorité de mon temps m'a toujours entouré, conseillé (sur la vie, les investissements à long terme, la mécanique... Etc.). Des amitiés s'y sont nouées, qui ne disparaîtront pas. Le département AMAP-CIRAD n'est pas en reste : merci à Yann Guédon pour ses cours de perfectionnement en statistiques, merci à Christophe Godin et Pascal Ferraro pour leur conseil et la précieuse aide qu'ils m'ont apporté pour les techniques d'appariement, et merci à Jean Dauzat et Christophe Nougier pour leur collaboration dans la représentation graphique de mes petits ballons ovales.

Toute cette puissante armée de chercheurs nécessite bien sûr un support logistique bien huilé, et je remercie chaleureusement Catherine Loubet (CIRAD), Odile Chouillou (CIRAD) et Marie-Line Barnéoud (ENSMSE) pour leur présence constante et leur gentillesse extrême dans tous les domaines. J'adresse un remerciement particulier à M. Pierre Igier qui a eu le courage de prendre en charge l'impression de ce mémoire de thèse, tâche complexe s'il en est. C'est promis, à ma prochaine thèse, je change de traitement de texte.

A présent vient le tour de ceux qui m'ont supporté (dans tous les sens du terme) et pour qui mon affection est sans borne : mon père, ma mère, mon frère Philippe. Je ne vous remercierai jamais assez de la chance que vous m'avez donné de pousser si loin mes études, ni de l'amour que vous me portez tous et sans lequel je ne serai rien. Merci à mes grands-parents, toujours curieux de savoir « où j'en suis », confiants en mes « capacités » qu'ils n'ont jamais mis en doute. Votre affection est mon plus grand bonheur. Des baisers d'amour pour Manel, qui a dû subir mes moments d'absence – et mes absences tout court - plus d'une fois, faisant montre d'une affection sans borne envers moi malgré les tumultes de la vie. Je t'aime Manel.

Merci à mes amis qui ont tant fait pour moi, toujours près de moi même si beaucoup suivent aujourd'hui leur propre route, avec les embûches que cela suppose. Vous êtes ma force et mon courage.

Enfin, j'adresse une pensée très forte à mon grand-père maternel à qui je dédie ce travail, lui qui prônait tant les études et qui est parti trop tôt. J'espère que tu nous vois aujourd'hui, car nous sommes, Philippe et moi, tout entier tourné vers toi. Nous voulons te dire combien nous t'aimons et que nous ferons de notre mieux pour suivre tes préceptes et respecter ta mémoire : tu es notre phare dans la nuit.



# 0. Introduction.

## 1. MOTIVATIONS.

Cette thèse d'informatique propose un modèle – une *représentation informatique* – pour structurer et caractériser les formes géométriques extraites d'*images tridimensionnelles*. Nous désignons par « image tridimensionnelle » un ensemble de points régulièrement positionnés selon trois dimensions et dotés de propriétés supplémentaires de couleur, densité ou toute autre information à caractère continu ou discret.

Le *traitement* des images 3D est difficile et coûteux en mémoire et en temps, phénomène aggravé par la résolution sans cesse croissante des modalités d'acquisition. Le terme « traitement » regroupe les opérations d'acquisition, de stockage, de transfert et enfin d'analyse.

Le contexte médical de ce travail définit les aspects de traitement comme suit :

- l'extraction d'informations pour le diagnostic médical : détection de fractures, de tumeurs, de déformations pathologiques, préparation de trajets d'instruments chirurgicaux... ;
- l'obtention de la surface des organes, cette donnée n'étant pas directement fournie par les images acquises ;
- la visualisation interactive ; il convient qu'elle mette en valeur les informations pertinentes pour le praticien ;
- la constitution d'un atlas anatomique, à des fins pédagogiques ou de simulation ;
- tout autre filtrage des données.

Ces traitements interviennent après une phase de segmentation – c'est-à-dire de détection des différents tissus – et sont impossibles à partir des images brutes. Il faut pour cela projeter les données en un modèle de plus haut niveau, qui extrait, synthétise et offre un accès contrôlé aux informations pertinentes. Nous proposons à cette fin le modèle de *E-Squelette* (ou *squelette ellipsoïdal*). Il constitue une méthode automatique de filtrage des données préalablement segmentées et dote ces dernières des fonctionnalités suivantes :

- visualisation 3D multi-échelle, autorisant la manipulation en temps réel des entités ;
- mesure de l'évolution structurelle des organes au cours du temps au travers des paramètres désirés ;
- détection de différences locales entre un élément donné et un élément de référence (détection de fractures par exemple) ;
- reconnaissance automatiques d'organes rigides à partir d'une bibliothèque de *E-Squelettes* de référence ;
- obtention automatique de la surface des organes par ajustement surfacique avec erreur contrôlée ;
- compression des données grâce à l'expression purement analytique de la géométrie.

## 2. CADRE DE TRAVAIL.

Le modèle de E-Squelette s'inscrit comme module de l'application *Corpus 2000* développée au CIRAD dans le département *Modélisation des Entités Biologiques*. Créé par MM. Marc Jaeger et Philippe Borianne, il traite les images 2D sériées d'origines médicale ou forestière acquises par scanners ou IRM. Une phase de segmentation est ensuite exécutée afin de détecter les tissus selon leurs propriétés de texture, densité... Etc. C'est en sortie de ce processus qu'intervient le E-Squelette de manière automatique : les ensembles de points définis comme appartenant à des entités distinctes vont subir une transformation permettant leur analyse et leur caractérisation.



Figure 1

L'application *Corpus 2000* avec le module de E-Squelette activé (fenêtre en bas à droite) : il autorise une vision dynamique tridimensionnelle de la zone concernée (en violet et jaune : foie et tumeurs, en bleu les reins). A tout instant un organe peut être sélectionné, étudié selon les critères désirés (volume, poids, ... etc.) et comparé à d'autres organes stockés en mémoire de masse.



### 3. PLAN DU RAPPORT.

Nous offrons tout d'abord dans le premier chapitre une récréation pour l'esprit curieux, puisqu'il s'agit d'un petit tour d'horizon de la réalité virtuelle. Le but de ce chapitre est de sensibiliser le lecteur aux problèmes de représentation. Nous verrons les problèmes qui surgissent lorsque l'on veut offrir à l'esprit humain une représentation totalement synthétique d'un univers, quel qu'il soit.

Les chapitres 2, 3 et 4 constituent des états de l'art des grands thèmes que nous avons traités au travers de notre modèle de E-Squelette. Le chapitre 2 présente comment les solides ou objets tridimensionnels numériques sont décrits aujourd'hui au sein de l'ordinateur. Le chapitre 3 répertorie quelques techniques pour reconstituer la surface des objets, autorisant ainsi leur vision et leur appréhension. Le chapitre 4 introduira le lecteur aux méthodes permettant la reconnaissance automatique de formes par la machine. Ces trois chapitres mettent en place les zones d'influence du E-Squelette.

Le chapitre 5 décrit le modèle E-Squelette et propose l'intégration de certaines méthodes dans celui-ci. Des expérimentations sont présentées, mettant en valeur la stabilité du modèle, condition *sine qua non* de la caractérisation d'entité.

Le chapitre 6 explore les capacités de visualisation, d'interaction et de reconstruction fine de la surface d'un solide tridimensionnel grâce au E-Squelette. Des techniques robustes sont présentées, ainsi que des démonstrations du fonctionnement de ce que nous nommons *visualisation dynamique*.

Le chapitre 7 montre l'essence même du E-Squelette, qui est de fournir une carte d'identité de chaque objet tridimensionnel. Nous montrerons comment l'identification automatique et la classification sont rendues possibles, et comment une distance peut être mesurée entre deux entités.

Nous terminerons ce cheminement par une ouverture sur les perspectives d'évolution du E-Squelette, qui peut et doit encore changer pour s'adapter à des champs d'application particuliers. De multiples voies ont été ouvertes, et nous discuterons de celles qui semblent présenter le plus de potentiel dans les domaines de la représentation, visualisation et reconnaissance de forme.

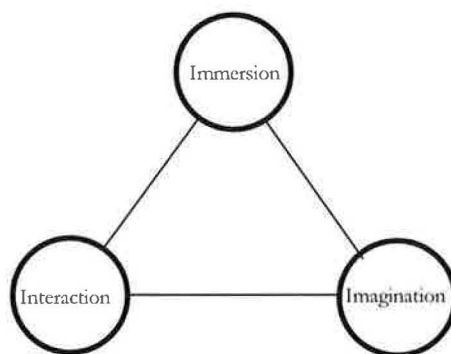
# 1. La Réalité Virtuelle.

## 1. INTRODUCTION.

Aujourd'hui très médiatisée, la Réalité Virtuelle semble, pour la plupart d'entre nous, être née dans les années 90. C'est essentiellement son ouverture au grand public qui en est la cause, et tout spécialement dans le domaine des loisirs. Comme nous allons le voir, le spectre d'application de la Réalité Virtuelle est très vaste, et son histoire un peu plus ancienne qu'il n'y paraît...

### 1.1. Qu'est-ce que la Réalité Virtuelle ?

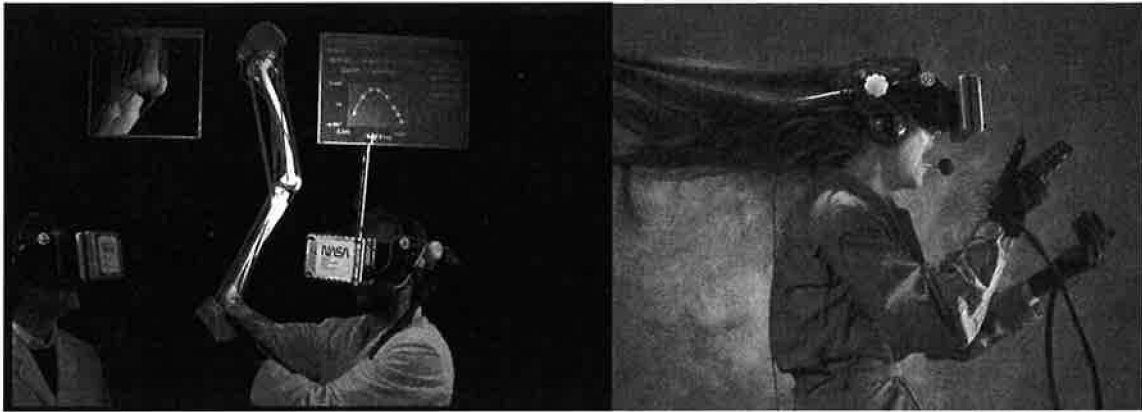
Le but recherché de la Réalité Virtuelle est l'immersion totale de l'être humain dans un univers, un espace totalement artificiel dans lequel il pourra évoluer et surtout interagir. Pour ce faire, des outils doivent être mis au point, qui auront pour fonction de « tromper » nos cinq sens et ainsi nous donner l'illusion d'une réalité « concrète », intuitivement semblable à celle que nous connaissons dans la vie de tous les jours. C'est la signification du fameux triptyque *Immersion, Interaction, Imagination* [20], que symbolise le schéma suivant :



*« Un système de réalité Virtuelle est une interface qui implique de la simulation en temps réel et des interactions via de multiples canaux sensoriels. Ces canaux sensoriels sont ceux de l'homme : vision, audition, toucher, odorat, goût. » [20]*

La Réalité Virtuelle revêt bien des visages, souvent très différents de ceux que présentent les médias. Nous allons en découvrir quelques uns en nous attachant aux problématiques qu'elle soulève. Enfin, nous nous focaliserons sur son utilisation en Imagerie Médicale, thème approprié pour introduire mon travail de thèse de doctorat.

Nous ne développerons pas en détail un aspect étendu de la réalité virtuelle, nommé *réalité augmentée*. Cette technique consiste à enrichir une vue réelle – par exemple en superposant des images de synthèse à une vue vidéo ou une plaque transparente – dans le but d'améliorer les perceptions ou de guider l'expérimentateur. Nous présenterons néanmoins certains dispositifs développant ce concept dans cette partie.



Quelques exemples de Réalité Virtuelle : un casque monté sur la tête, des gants et éventuellement une combinaison. L'image de synthèse du squelette de la jambe dans l'image de gauche nous indique ce que voient et manipulent les deux expérimentateurs.

(Courtoisie de NASA Ames Research Center (VIEWLab))

## 1.2. Un peu d'histoire...

Nous ne remonterons pas jusqu'en 1777, date à laquelle Charles Earl Stanhope inventa la première *machine logique*, bien que l'ordinateur d'aujourd'hui en dépende directement. Nous explorerons quelques événements depuis cette date, qui ont préparé puis jalonné l'histoire de la Réalité Virtuelle (RV).

Le pionnier de la RV fut sans conteste Morton Heilig – un professionnel du cinéma –, qui inventa en 1956 le *Sensorama*. C'était un appareil de cinéma destiné à « donner une illusion totale de la réalité avec une vision tridimensionnelle, des odeurs, du vent, des variations de températures, des sensations corporelles [...] » [Interview de M. Heilig réalisée en 1989].



Le *Sensorama*, de Morton Heilig.

Afin de convaincre les éventuels investisseurs, M. Heilig bâtit un prototype à une place du *Sensorama* en 1960. Le siège vibrat, des ventilateurs simulaient le vent, deux optiques fournissaient l'image tridimensionnelle et un dispositif pour les odeurs complétait le tout. Les images étaient imprimées sur deux pellicules 35 mm, les tubes cathodiques étant encore trop chers et complexes pour l'époque. Un des films (il y en avait quatre en tout) les plus attractifs proposait une course à moto à travers New-York, que M. Heilig décrit ainsi :

« [...] le siège vibrerait avec la moto, on sentait les odeurs, le vent sur le visage... Central Park avec l'odeur d'herbe fraîchement coupée, le Brooklyn Bridge, Times Square et ses pizzas, la Vème avenue, on avait la sensation physique d'être sur la moto et de pencher dans les virages. »

Quelques limitations existaient néanmoins par rapport à l'idée originale, à savoir l'absence de variation de température, le champ de vision réduit à 100° au lieu de 180° et l'absence d'inclinaison du siège, jugée trop dangereuse pour une utilisation grand public. M. Heilig, se heurtant au scepticisme des potentiels investisseurs, lança l'exploitation de sa machine en utilisant des fonds propres.

En 1957, M. Heilig dépose un brevet pour un appareil se fixant sur la tête et fournissant une vision binoculaire en couleur grâce à deux tubes cathodiques. En 1963, Ivan E. Sutherland publie sa thèse de doctorat intitulée « *Sketchpad: A Man-Machine Graphical Communication System.* » qui va ouvrir la voie de la Réalité Virtuelle telle que nous la connaissons aujourd'hui : créée via des images de synthèse produites par ordinateur. Ce qui vient de naître alors est ce qui nous semble aujourd'hui très classique : un outil de dialogue entre l'homme et la machine, restée jusqu'alors très peu intuitive et donc très peu exploitable.



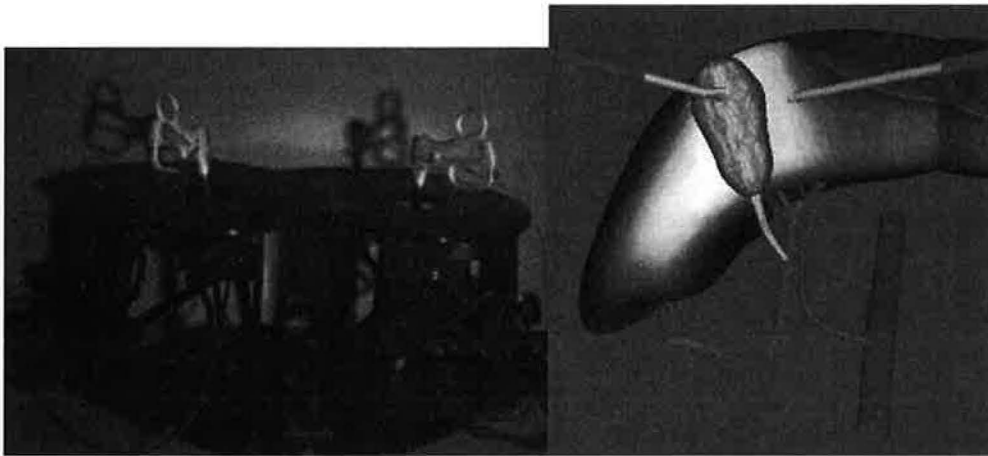
Deux ans plus tard, en 1965, I. E. Sutherland concrétise cette idée en reliant les tubes cathodiques à des calculateurs (des ordinateurs) qui vont générer un univers synthétique, immatériel. C'est dans sa fameuse publication scientifique « *The Ultimate Display.* » [18] qu'il va exposer son idée maîtresse :

« Don't think of that thing as a screen, think of it as a window through which one looks into a virtual world. The challenge to computer graphics is to make that virtual world look real, sound real, move and respond to interaction in real time, and even feel real. »

« N'assimilez pas [la RV] à un écran, considérez-la plutôt comme une fenêtre à travers laquelle vous regardez un monde virtuel. Le défi de l'image par ordinateur est de fabriquer des mondes virtuels qui semblent réels, qui répondent aux interactions en temps réel, que vous ressentiez comme réel. »

C'est cette phrase qui guide jusqu'à aujourd'hui les recherches concernant la Réalité Virtuelle.

Les années suivantes marqueront l'approfondissement et l'amélioration de ces concepts. En 1967, Frederick P. Brooks et son équipe vont produire dans le cadre du projet GROPE (de l'anglais *tâtonnement*) [21] plusieurs appareils dits *haptiques* : ils ajoutent une nouvelle dimension au monde virtuel, à savoir la sensation tactile.

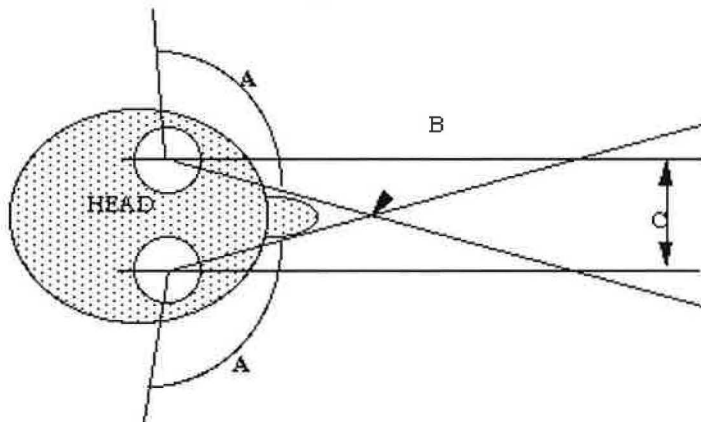


C'est Frederick P. Brooks qui jeta les bases des outils permettant la sensation tactile (*haptiques*). L'appareil moderne à gauche dispose de poignées pour les mains, rendant plus réaliste le travail réalisé à droite : l'image devient « solide ».

Bien que récent, ce dispositif s'inspire totalement des fondements posés par F. P. Brooks en 1967.

(Image de gauche courtoisie de Immersion Corporation, Image de droite courtoisie de Deutsches Zentrum für Luft und Raumfahrt)

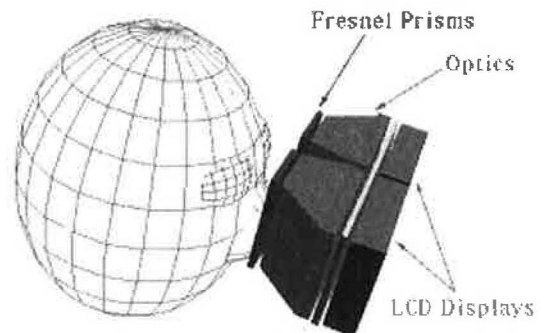
En 1968, I. E. Sutherland perfectionne encore son visionneur en modifiant le calculateur de telle façon que les deux images envoyées à chaque œil trompent le cerveau et donnent une réelle impression de volume à l'expérimentateur : c'est la *stéréoscopie*, une technique consistant à générer des images légèrement décalées d'une même scène pour que la troisième dimension apparaisse.



La **stéréoscopie** (du grec *stereo*, solide) :

Chaque œil, du fait de sa position (voir distance C), reçoit une image différente d'un même objet. Le cerveau fusionne les deux vues et le volume est naturellement recréé. Cette vision stéréoscopique fonctionne entre 10cm et 5 m environ. Au-delà, le système visuel emploie d'autres procédés en repérant des "indices" dans l'image : ombres, occultations...Etc.

Dans un affichage vidéo, il faut "recréer" mathématiquement cette variation : les deux écrans projettent donc une image particulière vers chacun des yeux de l'expérimentateur.

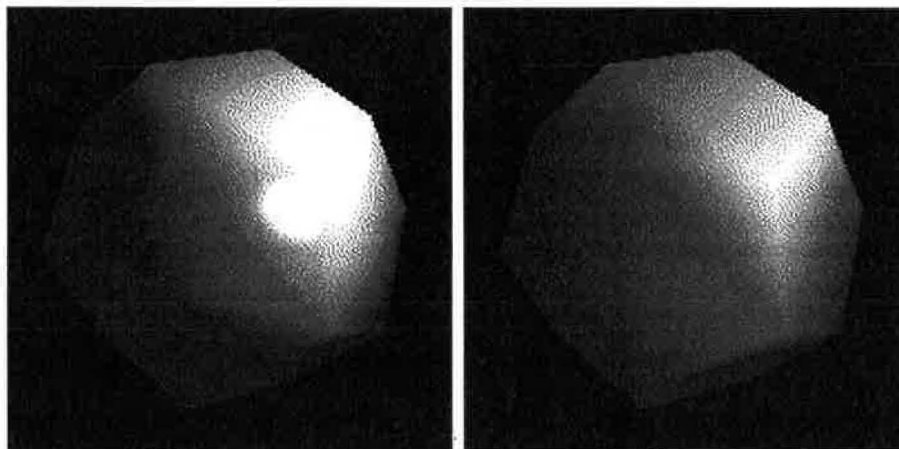


LCD HMD Components

Notons que le premier appareil de projection stéréoscopique fut construit en 1838 par le scientifique anglais Wheatstone.

A partir de 1970, l'armée américaine (*United States Army*) et la NASA (*National Aeronautics and Space Administration*) s'intéressent aux travaux de I. E. Sutherland à des fins de simulation et d'entraînement, et développent de nouveaux calculateurs plus performants.

Parallèlement, de grands progrès sont faits en ce qui concerne la qualité des images synthétiques produites par les calculateurs, grâce aux travaux de MM. Henri Gouraud [23] et Bui-Tuong Phong [22] sur l'éclairage des objets.



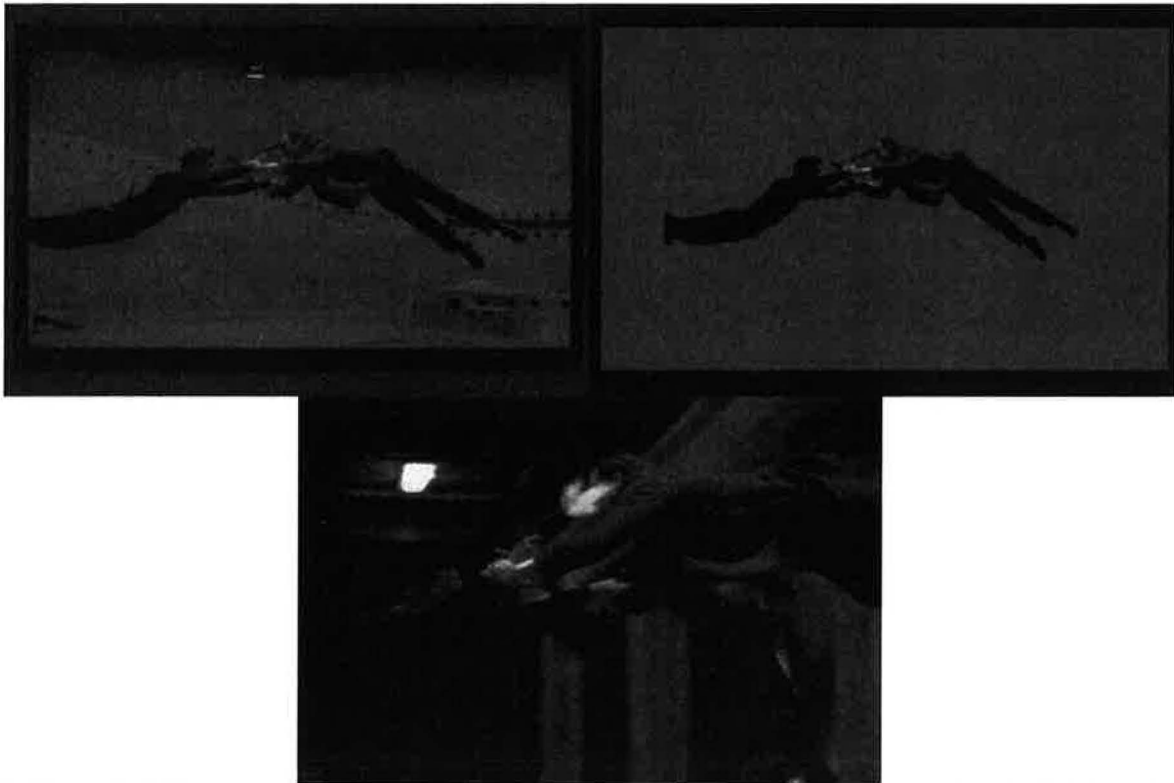
Des techniques particulières permettent d'améliorer la sensation de volume des objets : c'est l'*éclairage*, qui tente de rendre compte des ombres et de l'aspect réfléchissant des matériaux. Les techniques présentées ici sont respectivement dues de gauche à droite à B-T. Phong et H. Gouraud.



Bien qu'étant ici uniquement en deux dimensions, ce personnage semble être en relief grâce aux techniques d'éclairage décrites précédemment.

En 1983, Myron Kruger propose un système (*Mandala*) où l'on peut voir à l'écran (en deux dimensions seulement) sa propre personne mais plongée dans un univers graphique totalement artificiel. Il qualifiera ce procédé de « *Réalité Artificielle* », très utilisé à la télévision pour la météo par exemple, ou au cinéma pour incruster des personnages dans des lieux trop onéreux à construire, trop dangereux ou pour créer des situations extraordinaires :





L'ordinateur isole les personnage pendant l'action, en détectant les parties vertes (images gauche et droite). Il est alors en mesure de les superposer à n'importe quelle image de fond (image du bas). D'autres effets peuvent également être ajoutés comme le coup de feu (images courtoisie de Warner Bros, film *The Matrix*).

C'est William Gibson, auteur américain de science fiction, qui introduit en 1983 le terme de « *cyberspace* » dans son roman « *Neuromancien* » [24], un terme actuellement beaucoup – et fort mal – utilisé par les journalistes mais également les fabricants de jouets, d'ordinateurs...etc.

*« Le cyberspace. Une hallucination consensuelle vécue quotidiennement en toute légalité par des dizaines de millions d'opérateurs, dans tous les pays, par des gosses auxquels on enseigne les concepts mathématiques... Une représentation graphique de données extraites des mémoires de tous les ordinateurs du système humain. Une complexité impensable. »*

Le cyberspace selon W. Gibson.

1985 marque le début de l'ouverture de la Réalité Virtuelle à la « Réalité Economique », puisque la première entreprise, VPL Research Inc. est fondée aux Etats-Unis : c'est le premier pas vers le grand public. Le PDG de VPL Research, Jaron Lanier, consacre alors le terme de « Réalité Virtuelle ».

Les années 90 vont être le théâtre de l'explosion du concept de RV : commercialisation et introduction de la RV au grand public – notamment dans le secteur des loisirs -, création d'entreprises – qui suppose un accès de plus en plus banalisé aux outils d'immersion -, formalisation et consolidation des concepts et problématiques engendrées par la RV en recherche scientifique.





Voici un exemple de ce que l'on peut trouver dans le domaine du grand public pour la RV :

De gauche à droite : un volant qui réagit aux chocs et aux bosses de la route, des lunettes stéréoscopiques, un kit sonore pour percevoir le son dans toutes les directions et une manette de jeu rendant compte des tirs et chocs. La diversité et la richesse de l'offre montrent à quel point la RV devient une réalité économique de masse. Les concepts utilisés dans ces appareils ont plusieurs années, mais sont aujourd'hui pleinement abordables. (respectivement de gauche à droite : Images Courtoisies de Microsoft Corporation, ELSA, Creative Labs et Logitech)

Nous discuterons plus tard des potentialités futures de la RV. A présent, intéressons-nous aux principes qui régissent un tel concept, et essayons de comprendre pourquoi la RV « marche » ou « ne marche pas » !

## 2. L'IMMERSION, QU'EST-CE QUE C'EST ?

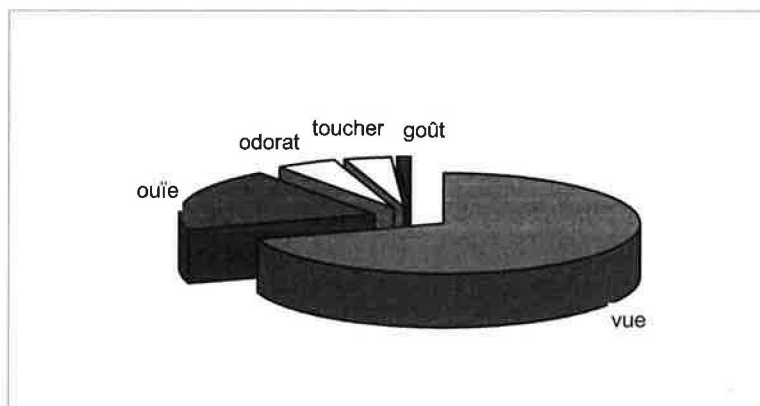
Nous avons vu que l'un des pivots de la RV est l'*immersion* de l'être humain dans un monde artificiel généré par l'ordinateur. Des phénomènes complexes sont à prendre en compte concernant notre façon d'appréhender l'environnement et les événements qui s'y déroulent. Fabriquer les outils adéquats qui tromperont nos sens passe inévitablement par l'étude du corps humain, dans ses aspects physiologiques et psychologiques.

A tout moment notre corps utilise et intègre deux types d'information : *sensorielles* et *proprioceptives*.

### 2.1. Les informations sensorielles.

Les informations sensorielles englobent la vue, l'ouïe, le toucher, l'odorat et le goût. Notons que le toucher englobe la notion de température, et pas seulement la sensation que procure un matériau selon sa *texture* (rugueuse, lisse, douce...etc.).

M. Heilig a proposé un diagramme qui rend compte – en moyenne – de l'importance de chacun des cinq sens dans la perception de l'environnement par le corps humain :



Le diagramme des perceptions selon Morton Heilig. Il indique l'importance que l'on doit accorder à chacune d'elle pour autoriser la sensation d'immersion.

Comme nous le verrons plus tard, le principal problème n'est pas seulement de stimuler chacun des sens mais surtout de fournir des combinaisons de stimulations qui semblent valides pour le cerveau.



Ce jeu (*The World is Not Enough*) propose au joueur d'évoluer dans un monde tridimensionnel. La richesse visuelle s'accompagne d'une gestion sonore efficace : avec le matériel adéquat, le son est parfaitement localisé dans l'espace (gauche, droite, centre mais aussi tout le champ arrière). Les jeux vidéo actuels intègrent tous la dimension sonore ainsi que le retour d'effort.

## 2.2. Les informations proprioceptives

Plus complexes à mettre en œuvre, elles méritent que l'on s'y attarde quelque peu. Lorsque nous évoluons dans le monde réel, notre corps reçoit constamment des informations sur sa configuration spatiale. L'oreille interne par exemple nous renseigne sur l'inclinaison et l'orientation, et nos muscles – par leurs tensions respectives – indiquent précisément la *posture* et les *mouvements* du corps. Ces données combinées sont envoyées au cerveau via le système nerveux sympathique : une « image mentale » de notre corps est obtenue. C'est ce phénomène perceptif que l'on nomme *proprioception*. Ce terme vient du grec « sensation » (*-ception*) et « corps » (*proprio-*).

Deux activités concrètes illustrant la proprioception :

- Faites une croix sur une feuille blanche. Stylo en main, élevez la feuille au-dessus de votre tête, fermez les yeux et tentez de tracer un point le plus près possible de la croix. Regardez le résultat et recommencez la même expérience plusieurs fois.
- Fermez les yeux et levez les deux mains au-dessus de votre tête. Tout en gardant la main gauche fixe et ouverte, touchez alternativement votre pouce gauche et votre nez grâce à l'index de la main droite. Répétez plusieurs fois le processus. Faites vous des progrès à chaque nouvelle tentative ?

Que se passe-t-il ?

Vous utilisez des capteurs –nommés *propriocepteurs* – pour obtenir une image mentale de votre corps et de sa configuration dans l'espace. Nous dépendons beaucoup du stimulus visuel, et la proprioception n'est pas suffisante pour nous donner une information fine sur

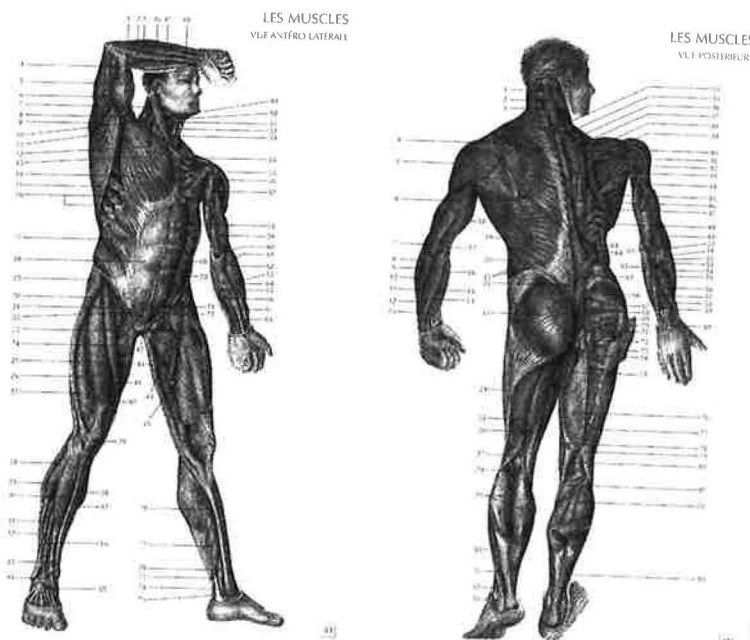
notre position. Les essais répétés des activités proposées permettent de corriger petit à petit l'image mentale et donc la précision.



Ce batteur de jazz utilise pleinement la proprioception : elle vient compléter l'information visuelle afin d'assurer régularité, vitesse et multiplicité de mouvements simultanés.

Les propriocepteurs sont placés dans les muscles, les tendons, les articulations et l'oreille interne (plus précisément l'*appareil vestibulaire*) et envoient des informations au *système nerveux central*. La peau contient également ce type de récepteurs, qui ne nous renseignent pas sur les sensations de toucher mais plutôt sur les tensions locales (étirement par exemple).

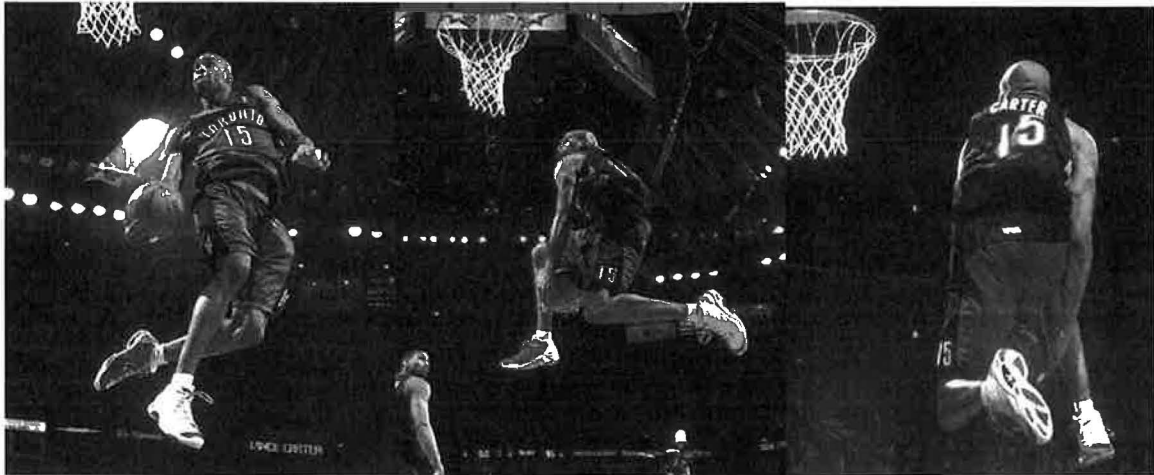
En anesthésiant certaines zones corporelles, il a été montré que le genou utilise peu ce dernier type d'information pour se positionner (muscles et articulations remplissent ce rôle) mais que la main y est particulièrement sensible.



Les muscles, tendons, ligaments, articulations ainsi que l'oreille interne et la peau fournissent un modèle mental du corps : les yeux fermés nous restons conscients de notre configuration corporelle. Cette capacité est la *proprioception*.

Il est intéressant de constater que ces informations proprioceptives peuvent être utilisées de deux manières par le cerveau :

- elles causent une réaction immédiate, comme par exemple déplacer tel ou tel membre pour retrouver son équilibre ;



Dans sa phase de saut, ce joueur de basket américain (Vince Carter, Toronto Raptors) utilise naturellement la proprioception pour s'équilibrer : ses jambes se coordonnent à l'information renvoyée par l'oreille interne, afin de lui donner une assiette stable. (images courtoisie de la National Basketball Association)

- elles sont enregistrées en mémoire, et seront réutilisées plus tard lorsque la situation sera semblable.

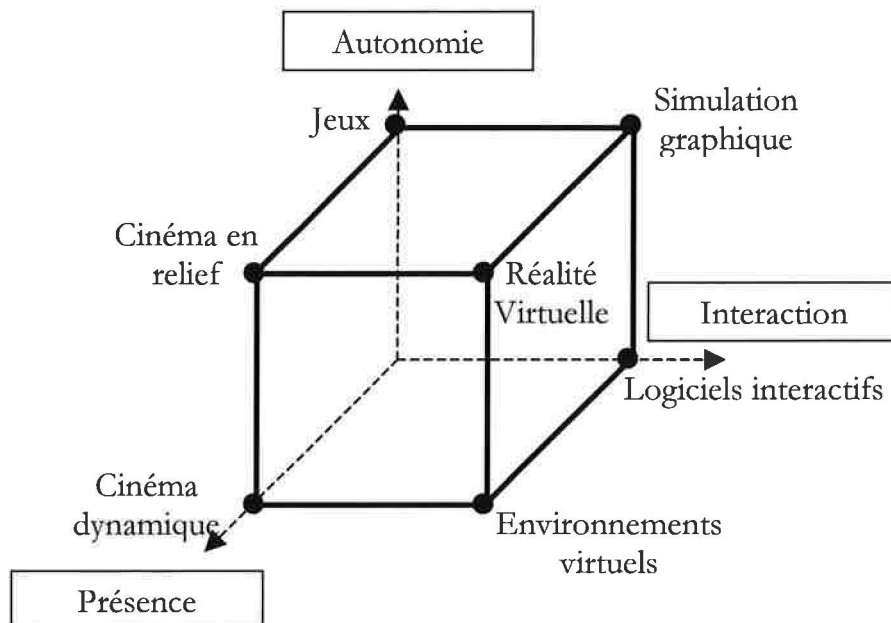
### **2.3. La Combinaison des Informations est la clé du succès.**

Pour rendre convaincante l'immersion dans un univers virtuel, le système générant les *stimuli* doit prendre en compte tous les paramètres que nous venons de présenter, et surtout rendre consistante la combinaison de ces paramètres (exemple du couplage audition-vision). Il semble néanmoins évident que la simulation doit intégrer bien plus que la simple somme des *stimuli* considérés séparément, ce qui soulève des problématiques complexes :

*« La multidimensionnalité et le nombre de paramètres qui commandent le comportement humain, auxquels s'ajoute la variabilité individuelle, interdisent tout modèle à prétention exhaustive ou définitive. »*

[1]

Le graphique suivant synthétise les trois grandeurs qui entrent en jeu dans la réalisation d'une telle tâche :

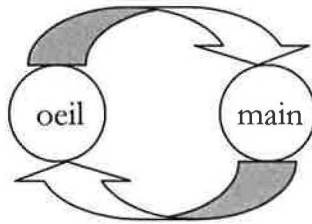


Le cube AIP (*Interaction/Autonomy/Presence*) de Zeltzer (*Massachusetts Institute of Technology*) montre la position des différentes applications informatiques selon les critères de Présence, d'Interactivité et d'Autonomie.

- Présence : qualité de traitement des canaux sensoriels en entrée ou en sortie ;
- Interaction : degré de contrôle de l'environnement virtuel proposé ;
- Autonomie : capacité des acteurs et objets du monde virtuel à agir selon le contexte et d'éventuelles loi physiques.

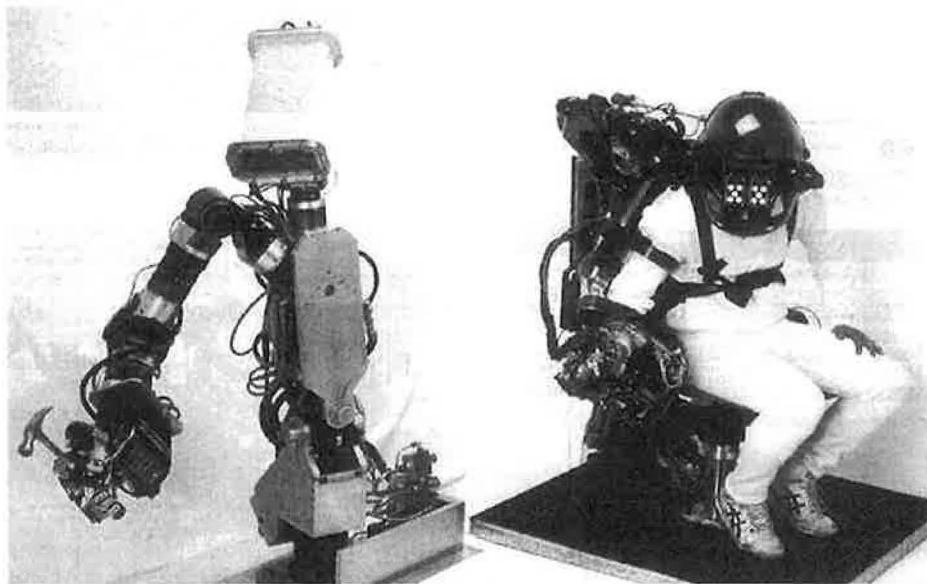
De multiples rétroactions surviennent dans le processus de réception des *stimuli*. Le Pr. Russ Neuman du *Media Lab* a prouvé par exemple que les gens percevaient une qualité visuelle bien meilleure lorsque la partie sonore était enrichie (spatialement ou qualitativement). Dans le même ordre d'idée, des prototypes de simulateurs de conduite de char d'assaut américains ont révélé que l'attention et l'efficacité des conducteurs était amplifiée en ajoutant simplement un dispositif produisant des vibrations semblables à celles dues au moteur.

La rétroaction constante au cours des interactions avec le milieu est clairement mise en évidence dans la coordination main-œil par exemple :



La coordination main-œil est un exemple frappant des subtiles relations entre canaux sensoriels : l'œil guide la main dans un premier temps. Pour rendre valide ce qui est perçu dans le monde virtuel, il faut que l'action de la main soit parfaitement répercutée au niveau visuel. La sensation d'immersion est améliorée, et par voie de conséquence l'efficacité du manipulateur.

Naturellement, le sens tactile ou *sens haptique* est un aspect incontournable du problème en ce sens qu'il guide les réactions corporelles. Des études menées par le professeur Raymond Goentz [1] ont montré que la technique de « *retour d'effort* » améliorerait grandement la dextérité, par exemple dans le contrôle à distance de robots :



Le manipulateur (à droite) contrôle le robot à distance (placé par exemple en environnement hostile). Les caméras du robot envoient une image stéréoscopique vers le casque du manipulateur. Pour autoriser une grande efficacité humaine, le robot doit renvoyer les résistances qu'il rencontre comme les obstacles ou la dureté des matériaux manipulés. C'est ce qu'on appelle le *retour d'effort*. De cette manière, l'opérateur est exactement dans la même situation que la machine et peut donc réagir au mieux en fonction du contexte.

Tout doit concourir à ce que le participant développe des comportements similaires à ceux qu'il aurait dans le monde réel. Pour cela les mouvements du corps entier doivent être pris en compte par le dispositif de RV, et les retours dans les canaux sensoriels doivent sembler prévisibles et cohérents. C'est par ce biais que la sensation de « présence » dans un monde synthétique sera obtenue.

Nous verrons plus avant les problèmes liés à la stimulation de nos sens, mais à présent il est temps de découvrir quelques outils actuels de la RV afin de rendre plus concrètes

toutes ces considérations mais également pour découvrir quelles sont les approches employées pour l'immersion.

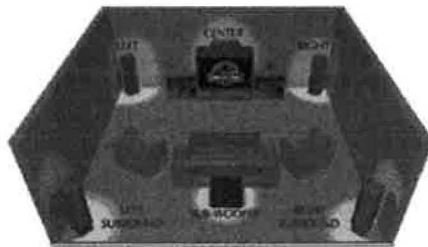


### 3. LES OUTILS DE L'IMMERSION.

Nous allons les découvrir en fonction du type de stimulus qu'ils apportent à l'utilisateur.

#### 3.1. Voir et Entendre.

Si l'on se souvient du diagramme de Morton Heilig, les sens de la vision et de l'audition sont prépondérants en tant que source d'information. Une grande attention est donc prêtée à ces canaux sensoriels. Avant d'examiner les produits, nous allons faire une analogie avec le cinéma actuel qui se pose également le problème de l'immersion :



Au cinéma, le son provient de l'avant (gauche, centre et droite) mais aussi de l'arrière (gauche, droite, ou ambiant). De cette manière le spectateur est « plongé » dans l'image. On notera qu'à la différence de la RV ce n'est pas lui qui agit sur ce milieu, c'est le réalisateur qui choisit où placer le spectateur.

(Image courtoisie de DTS)

Quant à la vision stéréoscopique, elle est techniquement gérée de la façon suivante :

- pour la vidéo, chaque écran projette des vues distinctes vers l'œil gauche et l'œil droit. C'est la fusion des deux images qui crée l'illusion de volume ;
- pour une vue extérieure – comme dans un cinéma par exemple – une même écran doit offrir deux vues différentes à chaque œil. Pour ce faire on emploie des *lunettes à obturateurs à cristaux liquides*, (LC : *Liquid Crystal*).



fusionnées par le cerveau.

Les lunettes à obturateurs LC utilisent le principe suivant : activés électriquement, les « verres » LC deviennent tantôt opaques, tantôt transparents. Chaque œil est ainsi alternativement masqué. Les lunettes sont synchronisées avec l'écran qui envoie alternativement l'image pour l'œil gauche et l'image pour l'œil droit. La vitesse de clignotement (de l'ordre de 60 à 100 fois par seconde) donne l'illusion de stabilité et les images sont naturellement

Voici quelques appareils spécialisés dans la vision :



Le ProView XL50 de Kaiser Electro-Optics possède deux écrans 1024x768 pixels en couleur et autorise la vision stéréoscopique. Il gère également le son stéréo (prix : environ US\$ 15000). Les écrans utilisés sont des matrices LCD (*Liquid Crystal Display*) de qualité élevée, ce qui explique en partie le prix élevé d'un tel matériel :



Cette matrice LCD fait 480 pixels sur 234 pixels (environ 5cm sur 7cm). Son prix est d'environ US\$320. C'est ce type d'écran qui est utilisé dans les lunettes de réalité virtuelle, car il est léger et facilement miniaturisable. Cependant son coût de fabrication est environ de trois à cinq fois plus élevé qu'un tube cathodique de même taille. (Image courtoisie de AEI Components).

Le ProView 50ST possède deux écrans 640x480. Sa particularité est d'autoriser la vision en transparence : l'image virtuelle peut être superposée à la vision réelle. (prix : environ US\$49000).



Le Lite Vision de LitEye ne gère pas le son. Il est employé pour le pilotage d'hélicoptères. Outre la vision nocturne, il peut apporter d'autres informations comme la vision infrarouge ou une vue extérieure (caméra externe). Ce n'est pas de la réalité virtuelle à proprement parler, mais c'est un exemple d'utilisation de ce concept pour « enrichir » la réalité. On nomme cela réalité augmentée car elle augmente les perceptions et la richesse des informations qui parviennent à l'expérimentateur.

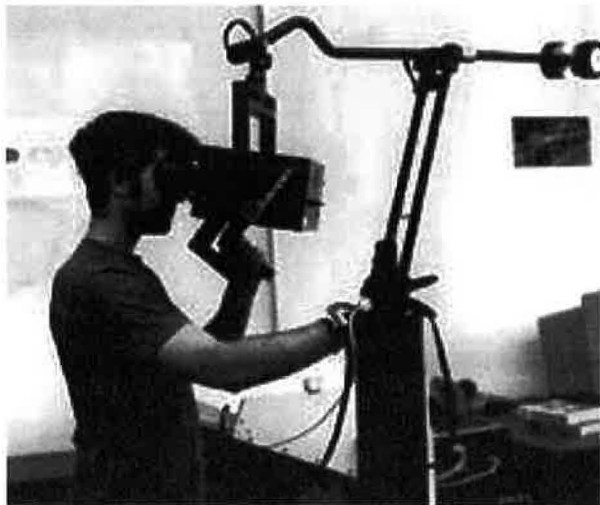


Vision infrarouge : ce ,n'est pas de la RV, mais c'est néanmoins une vision artificielle du monde extérieur.

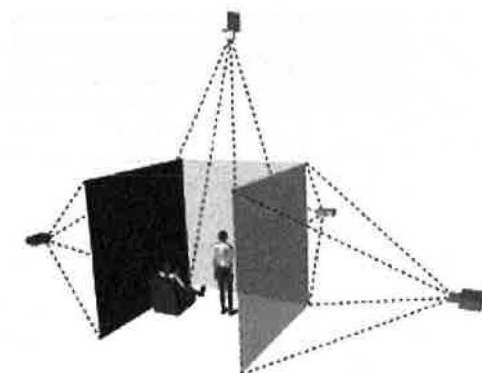


Le Dextroscope de KRDL est un *bureau virtuel* (virtual desktop). L'image artificielle est affichée sur une surface transparente. L'utilisateur perçoit le relief grâce à des lunettes à obturateur LCD. En transparence il peut voir ses propres mains qui semblent « utiliser » les objets virtuels. Ce n'est pas un système d'immersion complète, mais il intègre beaucoup de concepts de la RV.

L'*ImmersaDesk* de Fakespace est un autre modèle de bureau virtuel. On notera la structure en bois de l'appareil, ceci afin de ne pas interférer avec les capteurs de position des mains qui utilisent un champ magnétique.



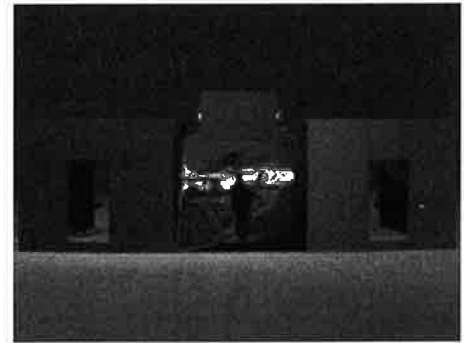
Le *BOOM* (*Binocular Omni-Orientation Monitor*: Moniteur Orientable Binoculaire) permet l'utilisation de tubes cathodiques de qualité. Le dispositif est monté sur un bras hydraulique qui permet de s'affranchir du poids des écrans. La mobilité autour d'un tel appareil reste néanmoins très limitée. (Image courtoisie de Fakespace)



Le *CAVE* (Cave Automatic Virtual Environment) est une pièce où des projecteurs vidéos envoient l'image sur plusieurs murs autour de l'utilisateur. Le son tridimensionnel est géré de la même manière que le cinéma, des lunettes à obturateur assurent une vision stéréoscopique, et un capteur détecte le mouvement de l'utilisateur pour recalculer l'image autour de lui. Ce système offre une

immersion de très bonne qualité notamment parce qu'il remplit totalement le champ visuel. (Image courtoisie de Fakespace).

On aperçoit l'utilisateur dans le dispositif CAVE : les murs qui l'entourent offrent des images qui sont recalculées constamment selon sa position.



### 3.2. Toucher.



Le *CyberGrasp* de *Virtual Technologies* agit sur la main pour simuler les objets solides : un système hydraulique retient les doigts de manière contrôlée selon la « dureté » de l'objet virtuel manipulé (environ US\$ 28000).

Le *CyberGlove* quant à lui simule la sensation de toucher en gonflant de petits ballons plats situés aux endroits-clé de la main (environ US\$ 15000).



Le gant est doté de capteurs internes et externes qui mesurent la position de la main et des doigts qui est ensuite répercutée dans le monde virtuel.





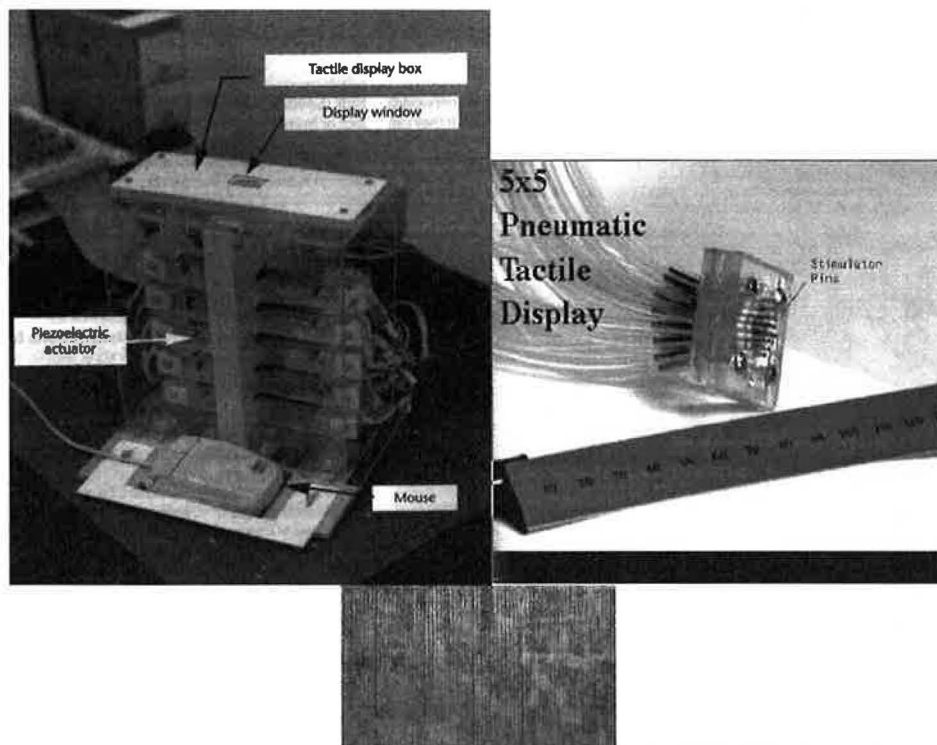
Le *Phantom* de *Sensable Technologies* autorise la manipulation d'objets virtuels tout en faisant ressentir à l'utilisateur les propriétés physiques de ceux-ci. Cet appareil est employé en RV médicale, comme nous le verrons plus tard. Ce type d'appareil entre dans la famille dite des *exosquelettes*. (squelette externe).

L'exemple typique d'utilisation du Phantom est la sculpture :

Le sculpteur travaille son bloc de matériau virtuel aussi intuitivement qu'il le ferait dans la

réalité.

Températures et textures restent encore très difficiles à simuler par un instrument mécanique. Certains appareils existent et utilisent des plaques à effet Peltier pouvant atteindre une température de 65°C. Pour la texture, des prototypes existent, comme celui-ci :



Cette machine (à gauche) est dotée d'une matrice d'aiguilles vibrantes de 5x10 (à droite), qui est capable de reproduire une sensation tactile à partir d'une image de surface (en bas). Le système s'inspire de l'*Optacon*, réalisé en 1960 pour les non-voyants. Les aiguilles vibrent piézoélectriquement selon les paramètres calculés à partir de l'image et fournissent l'illusion suffisante pour la peau [43].



La *Intensor Virtual Reality Chair* permet de ressentir les vibrations. C'est un outil très spécifique mais qui peut améliorer la sensation d'immersion. (Prix : environ US\$ 399). Ce type de produit est orienté grand public et montre l'émergence d'un nouveau marché dans le domaine des loisirs.

Le principal problème pour les stimulateurs haptiques concerne celui de la mise à jour des sensations. Tout comme un tube cathodique, l'effet doit sembler « constant » à l'utilisateur. On estime que la fréquence appropriée est de l'ordre de 250 Hz (250 fois par seconde) pour une réception cutanée maximale. [43].

### **3.3. Se Déplacer.**

Se déplacer suppose trois points importants :

- des capteurs de position pour que l'ordinateur puisse situer l'utilisateur dans le contexte virtuel, et ce sans aucun temps de retard ;
- pas de gêne dans le déplacement (fils, BOOM, ...etc.) ;
- pas ou peu d'interférences comme par exemple les capteurs magnétiques qui sont sensibles aux structures métalliques.

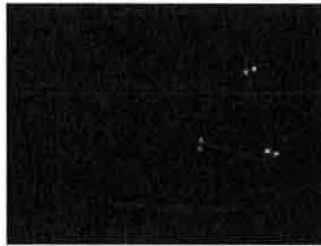


Les capteurs magnétiques *StarTrak* de *Polhemus* se fixent sur le corps (jusqu'à 16 capteurs par personne). Le modèle présenté ici ne requiert aucun fil entre l'ordinateur chargé d'enregistrer les mouvements (image de droite) et les capteurs. Les utilisateurs recouverts de capteurs évoluent dans un champ magnétique. Chaque capteur, sensible à ce champ magnétique, détermine sa position dans l'espace, et envoie ses coordonnées à l'ordinateur via un câble ou des ondes radio. La précision des capteurs est de l'ordre de 2 cm en position, et de  $0.75^\circ$  en orientation.



L'*IS600* de *Intersense* est basé sur la technologie ultrasonore. Les capteurs détectent leur orientation par un dispositif gyroscopique intégré dans chacun d'eux, et leur position par émission d'ultrasons vers une antenne en X (calcul du temps de trajet de l'onde). En

utilisant plusieurs antennes (et donc plusieurs ordinateurs de traitement) il est possible d'approcher la précision millimétrique.



Des détecteurs visuels peuvent également être employés. Ici les capteurs – placé sur un bras mobile pour la démonstration dans l'image de gauche – émettent dans le proche infra-rouge. Une caméra spéciale détecte les émissions (image de droite)

et en déduit la position. Pour reconstituer les trois dimensions, plusieurs caméras sont employées. (Image courtoisie de Dalhousie University, Canada). Des problèmes peuvent surgir par exemple lorsqu'un capteur est masqué durant le mouvement. L'ordinateur doit alors faire des hypothèses sur la position du capteur.

Chaque type de capteur (optique, magnétique, radio...) répond à des besoins particuliers, mais c'est au prix de contraintes spécifiques : les structures métalliques perturbent les champs magnétiques, les capteurs IR peuvent être masqués...etc. Les précisions sont également variables. Si la transmission n'est pas perturbée, il reste encore à l'ordinateur à gérer les données de manière très rapide : le monde virtuel doit « réagir » quasiment instantanément sous peine de perte d'efficacité, voire de malaise (voir la section 4). On estime que le délai de réaction de l'ordinateur doit être en dessous de 250 ms [17].

Pour se déplacer dans le monde virtuel, plusieurs approches sont adoptées :



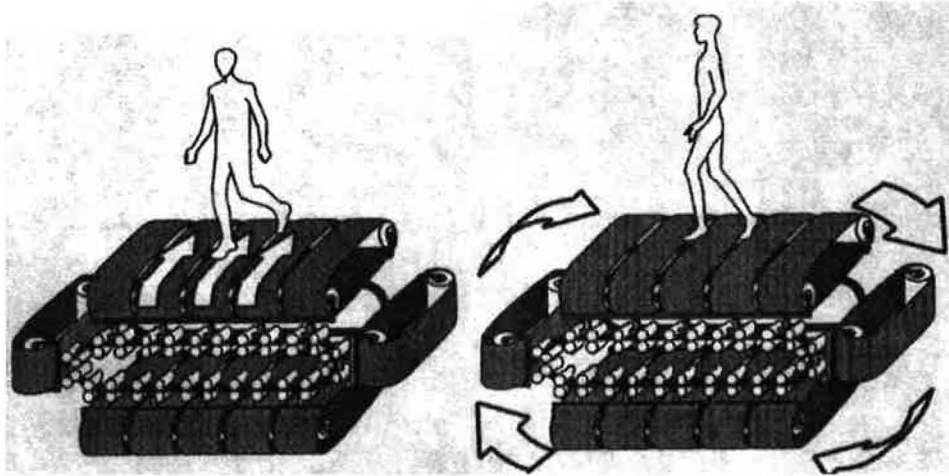
Le *Cyberstick* est une manette flottante (non rattachée à un socle) qui permet de se déplacer grâce à des touches de directions (à gauche). D'autres modèles permettent le déplacement par simple inclinaison de l'appareil. Dans ce cas aucun déplacement physique n'est effectué : c'est par action manuelle que l'on avance dans le monde virtuel. (Prix : environ US\$ 150). A droite, une vue plus précise du dispositif de déplacement sur une modèle *wand* pour le système *CAVE*



(image courtoisie de Fakespace).



La figure suivante nous montre un prototype de tapis roulant omnidirectionnel :



Ce tapis autorise le déplacement dans toutes les directions. [14]



#### **4. ET LE MAL DE MER DANS TOUT ÇA ?**

Aucun rapport me direz-vous ? Ce n'est pas si sûr... La RV peut induire des troubles pendant et après l'expérimentation. Ce phénomène n'est pas vraiment nouveau, puisqu'en 1957, bien avant toutes les technologies actuelles, des effets secondaires « indésirables » avaient été constatés par l'armée américaine dans les simulateurs d'hélicoptères [25].

Depuis, plusieurs expériences ont démontré que l'immersion pouvait produire des conflits inter-sensoriels chez l'homme, en observant par exemple des pertes de coordination main-œil lors du retour au monde réel.

##### **4.1. Ce que l'on a observé...**

Ce problème met en exergue la difficulté de régler et de calibrer les systèmes de RV afin que le corps puisse réagir comme s'il s'agissait de la réalité. Si les paramètres ne sont pas cohérents voici ce qui peut se passer :

- 1) L'expérimentateur met son équipement de RV, il est dans le « monde réel ».
- 2) L'expérimentateur s'accommode au système de RV et peut ensuite évoluer de manière satisfaisante dans le monde synthétique.
- 3) L'expérimentateur enlève son équipement de RV et revient au « monde réel ».

L'étape cruciale est l'étape 2, durant laquelle la personne doit s'accommoder au système et apprendre à réagir en fonction des stimuli reçus. « Notre organisme possède une très grande capacité d'adaptation dans ce domaine » comme l'explique le Dr. Rolland, chercheur à l'université de Caroline du Nord à Chapel Hill. Le problème survient lorsque le système RV est « mal réglé » : le corps apprend alors naturellement à compenser les « défauts » du système, mais lorsqu'il revient au monde réel, ses réactions deviennent aberrantes durant une certaine période – assez courte cependant.

Un exemple simple est de considérer deux voitures, l'une sans direction assistée, et une avec direction assistée. Faites l'expérience suivante :

- 1) Allez dans votre supermarché préféré aux grandes heures d'affluence, et gardez-vous avec la voiture sans direction assistée.
- 2) Retournez le lendemain au supermarché, dans les mêmes conditions et gardez-vous avec la voiture *avec* direction assistée. Faites cela pendant plusieurs jours.
- 3) Repartez encore une fois mais avec la voiture sans direction assistée...

Dans l'étape 2, l'organisme connaît une période dite de *calibration*, où il va s'accommoder au comportement de la direction assistée. Petit à petit l'aisance est acquise, et l'habitude s'installe. C'est ce que nous pourrions appeler la « période d'immersion ». Le problème survient à l'étape 3, où la réalité reprend le dessus, et la conduite devient alors pénible – heureusement durant une brève période.

Ce phénomène de changement brutal avait déjà été observé par les Romains, lorsqu'ils avaient remarqué combien les légions étaient perturbées – et donc beaucoup moins efficaces au combat – après de longs voyages en bateau. Habitué au tangage et au roulis sur une longue période, les soldats devaient subir un bref moment de réadaptation une fois sur la terre ferme pour recouvrer toutes leurs capacités.

Le fait de plonger l'organisme dans des conditions artificielles – ou tout au moins non adaptées - peut avoir des changements beaucoup plus subtils comme l'a observé le Dr. Reeves, à Stanford University à propos de changements de rythme cardiaque ou d'activité électrique dans le cerveau.

#### **4.2. Quels sont les symptômes ?**

Si l'exemple de la voiture ne produit pas d'état nauséeux, il n'en est pas de même pour les expériences de RV. Les symptômes les plus fréquents relevés par l'*US Army Research Institute* sont les suivants [19] (classés par ordre croissant de gravité) :

- malaise, inconfort
- maux de tête
- perte du sens de l'orientation ou de l'équilibre (*ataxie*)
- fatigue
- transpiration, salivation abusive, maux d'estomac
- nausées, vomissements.

Certains symptômes surgissaient longtemps après l'expérience par simple réminiscence du sujet. De manière générale, plusieurs symptômes sont observés, soit en totalité soit partiellement, ou aucun. Cette caractéristique polysymptomatique rend l'analyse très complexe [26].

Les symptômes observés sont identiques à ceux survenant dans le « mal de mer » ou le « mal des transports », même si parfois aucun mouvement physique réel n'est opéré. Ceci suggère que le « mal du simulateur » est principalement d'origine visuelle. Comme nous allons le voir maintenant, les causes ne sont pas aussi simples à identifier qu'il n'y paraît.

#### **4.3. Quelles sont les causes ?**

Cette citation [1] illustre bien les difficultés que rencontrent les chercheurs à cerner le phénomène :

*« [...] l'analyse pour savoir si une situation engendrera des nausées dépend d'une compréhension adéquate des mécanismes sur lesquels s'appuient le contrôle et l'appréciation de l'orientation et du mouvement. Les situations [...] demandent une compréhension de ces mécanismes au-delà de la limite connue et toute prédiction est fragile. » [1]*

De nombreux organismes de recherche travaillent à la mise au point d'un protocole non seulement pour évaluer les causes du mal du simulateur, mais aussi et surtout pour les prévenir. C'est le rôle du fameux *SSQ*, ou « *Simulator Sickness Questionnaire* » [27]. Ce questionnaire identifie 16 symptômes et les niveaux d'importance gradués de 0 (absent) à 3 (grave) ressentis par les expérimentateurs sur chacun d'eux. D'autres tests viennent enrichir ces résultats, et un suivi temporel est réalisé.

En ce qui concerne les causes, deux théories récentes (1986 et 1992) cohabitent, chacune ayant ses points forts et ses points faibles. Nous nous contenterons de les citer ici, sans toutefois prendre part à la polémique.

- la théorie du *conflit sensoriel* ou *cue conflict theory* : elle met en exergue le manque de consistance entre les perceptions. Par exemple, lorsque l'œil perçoit un mouvement alors que l'oreille interne n'enregistre aucune différence d'orientation. C'est le cas typique du simulateur de vol immobile. Le phénomène se produit aussi lorsque le mouvement réel – pour un simulateur sur plateforme mobile – ne correspond pas au mouvement perçu par l'œil (amplitude réelle et apparente). [25][28]. Un tel conflit produirait une réaction nauséuse. Selon M. Treisman [29], il s'agirait d'une tentative de protection de l'organisme qui associerait les conflits à une déficience des organes sensoriels dus à l'ingestion de toxines. Il s'ensuivrait alors une réaction de rejet (transpiration, vomissements) pour évacuer la substance incriminée.
- la théorie de l'*instabilité de posture* ou *postural instability* : c'est la théorie la plus acceptée aujourd'hui, qui prend comme hypothèse que la réaction nauséuse est entretenue tant que l'individu n'a pas de stratégie pour maintenir une position stable [30][31].

Nous allons présenter brièvement les facteurs associés au mal du simulateur, et montrer qu'ils sont de trois ordres, à savoir :

- les prédispositions individuelles (comme être sujet à l'épilepsie par exemple) ;
- la qualité du matériel utilisé ;
- la nature de l'activité à réaliser dans le monde virtuel.

#### 4.3.1. L'Individu.

Les aspects sont les suivants :

- l'âge, à savoir que les prédispositions sont plus grandes entre 2 et 12 ans, décroissent rapidement entre 12 et 21 ans et sont presque inexistantes à 50 ans [32] ;
- le niveau de concentration qui, s'il est élevé, réduit les possibilités de malaise [33] ;
- l'origine ethnique. Par exemple, les orientaux sont plus sensibles au mal du simulateur que les occidentaux, probablement à cause de différences génétiques dans les

sécrétions de catécholamine (neuro-transmetteurs, entre autre responsables de la proprioception) [34] ;

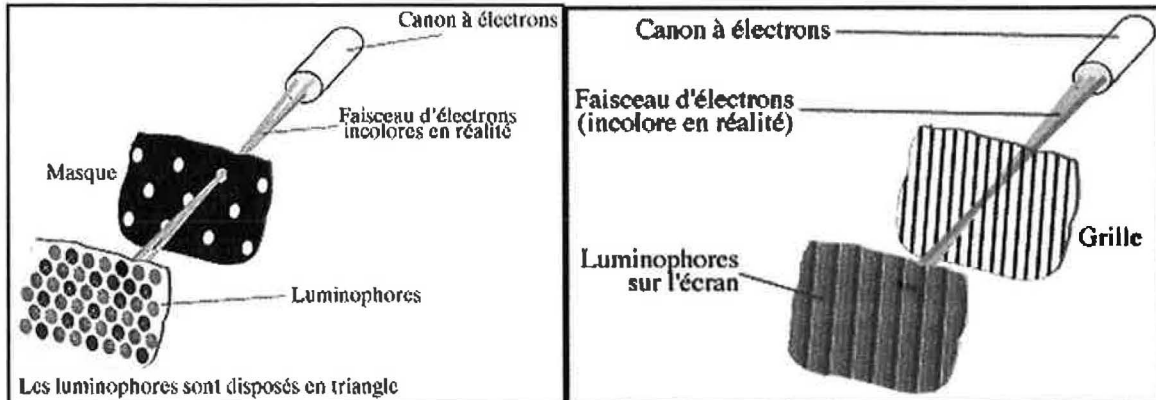
- l'expérience dans le « monde réel ». Plus cette expérience est importante, moins sont élevées les chances de malaises [35] ;
- l'expérience avec les simulateurs. Les sujets ayant subi des effets secondaires s'habituent progressivement aux simulateurs[36]. Des expériences suggèrent également que les problèmes de retour à la réalité augmentent parallèlement [37] ;
- la sensibilité propre au clignotement. Certaines personnes perçoivent le clignotement des écrans avec plus d'acuité. Cette sensibilité varie également entre le jour (maximale) et la nuit (minimale) ;
- le sexe. Les femmes ont généralement un champ de vision plus large que celui des hommes, ce qui les rend plus sensibles à la qualité du matériel employé ;
- les maladies ou états passagers. La fatigue, le manque de sommeil, le rhume...etc. augmentent les risques d'état nauséux [35] ;
- l'aptitude à la *rotation mentale*. Cette capacité, un peu particulière, exprime la faculté qu'ont certaines personnes de reconnaître des objets qui sont vus sous des angles très peu courants. D'après des études réalisées sur des cosmonautes soviétiques [38], cette faculté permettrait à l'expérimentateur de mieux compenser les conflits sensoriels comme avancer dans le monde virtuel sans se déplacer physiquement dans la réalité ;
- la stabilité physique. Les personnes exhibant une grande stabilité de posture en phase de pré-simulation sont statistiquement moins sujettes au mal du simulateur, selon des études menées par l'*US Navy*.

#### 4.3.2. Le Simulateur en lui-même (le matériel mis en œuvre).

Les outils d'immersion que nous avons découverts précédemment sont de qualités variables, essentiellement liées aux coûts de fabrication mais également aux contraintes de l'applicatif (légèreté, robustesse...etc.) et non à la technologie qui n'est pas forcément une limite en elle-même. Selon les prix, ces outils peuvent exhiber des faiblesses qui contribuent grandement au malaise en *per* (pendant) ou *post-simulation*. Nous en présentons ici quelques uns :

- la vision binoculaire : comme nous l'avons vu précédemment, la vision peut être soit monoculaire (un seul œil), ou binoculaire. Dans ce dernier cas l'image peut être identique pour chaque œil, on parle alors d'affichage *monoscopique*, ou différente pour percevoir le volume (affichage stéréoscopique). Des études ont montré que la vision stéréoscopique pouvait être source de malaise [39] ;
- champ de vision : l'ouverture du champ rend les sujets plus sensibles, en ce sens qu'il augmente les impressions de *vection*, et qu'il peut augmenter la perception de clignotement des écrans de visualisation. La *vection* désigne l'illusion de déplacement que l'on peut ressentir par simple stimulus visuel ;
- clignotement : les écrans vidéo doivent constamment mettre à jour les images pour qu'elles semblent stables. Si ce procédé est trop lent, l'œil perçoit le clignotement, ce

qui augmente le risque d'effets secondaires. Cette vitesse de mise à jour est appelée *taux de rafraîchissement*. Le clignotement quant à lui est davantage perçu par la vision périphérique, qui est plus sensible à ce phénomène que la vision fovéale [40] ;



Fonctionnement d'un écran : un faisceau d'électrons bombarde très rapidement des points de phosphore (*photophores*). Ces points restent allumés un certain temps (c'est la *rémanence*) sous l'effet de l'excitation. Si la vitesse de bombardement électronique est trop faible, l'œil perçoit le clignotement. (Image courtoisie de Zdnet).

- plateforme mobile : pour réduire les conflits sensoriels, certains simulateurs de RV sont dotés de plateformes mobiles. Des études [41] ont montré que cela ne résolvait pas tous les problèmes du mal du simulateur, et que les expérimentateurs pouvaient également ressentir des problèmes, tels que l'ataxie ;
- rémanence : sur les plaques LCD, le cristal a besoin d'un certain temps pour changer d'état. Si ce temps est trop long, l'œil peut percevoir des « traînées » sur l'écran. Il en est de même pour les photophores de tubes cathodiques ;
- détection et calcul de position : l'ordinateur utilise des capteurs (voir section 3.3) pour repérer le sujet dans l'espace réel. Il transforme ensuite ces données pour déterminer où se trouve virtuellement la personne dans l'espace synthétique. Une erreur de précision peut résulter en des conflits sensoriels dus par exemple à la différence entre la position de la main dans le monde virtuel et la position réelle ressentie par proprioception. Les capteurs actuels présentent toujours de faibles distorsions dans ce calcul de positionnement. Une autre phénomène est lié à la vitesse de détection de la position : si celle-ci n'est pas suffisante, un effet de « retard » peut survenir, ce qui est une source importante de malaise. Un dernier obstacle concerne la sensibilité des capteurs : si ceux-ci sont très précis, ils risquent de se produire des effets de « tremblement » qui vont déstabiliser le sujet.

#### 4.3.3. La tâche à réaliser dans le monde virtuel.

- degré de contrôle : si l'interaction est grande dans la simulation, les risques sont réduits. Il semblerait que la capacité de contrôle accrue permette au sujet d'anticiper plus facilement les réactions du « monde virtuel », et donc de réduire les conflits sensoriels ;

- durée d'immersion : l'incidence des effets secondaires et le temps d'adaptation lors du retour au monde réel augmentent avec la durée de la simulation ;
- valeur du rapport entre la vitesse de mouvement et la hauteur de l'œil : en haute altitude, les troubles ne se font plus sentir. Il en est de même pour la vitesse de mouvement du sujet, qui doit être inversement proportionnelle à la hauteur perçue. D'où l'intérêt de minimiser ce rapport appelé *flot visuel global* ;
- *vection* : la perception du mouvement induite par le stimulus visuel peut être source de malaise.

#### 4.3.4. Conclusion

Comme nous avons pu le voir, il existe une grande variété de causes du mal du simulateur. C'est semble-t-il le plus souvent une combinaison de ces causes qui va conduire le sujet vers des troubles. Facteurs individuels et techniques se mêlent sans qu'il soit pour l'instant possible de les dissocier. Des recherches dans ce domaine concernant la mise au point de protocoles de prédiction – comme le SSQ – semblent indispensables.

## 5. LA RV ET LE MONDE MÉDICAL.

Ces dix dernières années ont marqué des changements significatifs dans le domaine de l'imagerie médicale, fusionnant plusieurs champs d'expertise restés jusqu'alors disjoints : la médecine, le traitement d'images, la synthèse d'images et la robotique. Nous allons essayer de planter le décor de ce théâtre complexe : Que peut apporter la RV à la médecine ? Quels sont les impératifs et limites imposés dans ce contexte ? Ces questions seront suivies d'une présentation de différents matériels et applications à disposition des praticiens, qui comme nous le verrons, sont le plus souvent encore à l'état de prototypes.

### 5.1. Les apports de la RV.

Depuis la découverte des rayons X par Roentgen, les outils de vision intra-corporelle se sont multipliés, comme les scanners ou les IRM (Imagerie par Résonance Magnétique) avec une progressive baisse des coûts. La flexibilité des appareils d'acquisition autorise à présent la capture et l'analyse d'images en mouvement, intégrant la dimension temporelle.

En 1966, Isaac Asimov publiait « *Fantastic Voyage* » [44] où des chercheurs sont réduits à l'échelle microscopique pour être injectés dans le corps d'un scientifique qu'ils doivent sauver. Ce roman illustre cette nécessité qu'ont les praticiens d'explorer, d'expérimenter et de travailler dans le corps humain à des fins de diagnostics, de thérapie ou de recherche. L'outil de RV offre de nouvelles possibilités d'investigations et d'interaction, que nous allons essayer d'identifier :

- supprimer ou limiter les opérations invasives, comme c'est le cas par exemple de la *coloscopie* virtuelle (exploration du colon après examen scanner) ;
- mieux entraîner les jeunes praticiens – dont la formation requiert un grand nombre d'années – et notamment sur des cas peu courants. La RV permet de répéter à l'infini des cas particuliers d'opération par exemple, ou de définir des scénarii d'intervention ;
- baisser les coûts de formation. L'apprentissage progressif sur des patients peut entraîner une augmentation des temps d'occupation des salles d'opération par exemple et donc induire des surcoûts pour l'établissement hospitalier ;
- réduire les risques sur les patients lors de la formation, dans le cas par exemple d'un simulateur d'anesthésie ;
- mieux présenter les données (anatomiques, physiologiques...etc.) aux élèves mais également aux patients pour leur expliquer ce qui va se passer en phase pré ou post opératoire ; la RV présente un très bon support de discussion pour les phases de préparation ou de *briefing/debriefing* ;
- autoriser la téléopération, c'est-à-dire l'opération à distance des patients, sans que la présence du chirurgien spécialisé soit requise.

Toutes ces raisons font que la RV médicale devient un axe de recherche sans cesse plus important. Il faut bien comprendre que certaines applications ne sont encore qu'à l'état d'expérimentation et qu'il peut encore s'écouler de nombreuses années avant leur

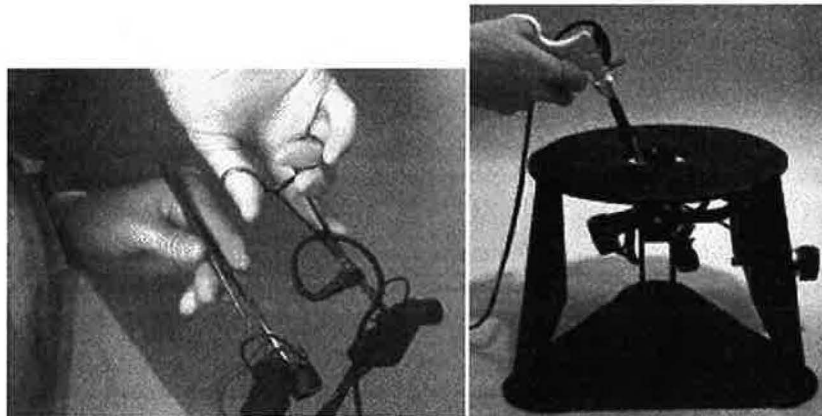


utilisation effective. Néanmoins, une voie est ouverte, et les recherches actuelles feront les outils de demain.

## **5.2. Les impératifs de l'imagerie médicale**

L'imagerie médicale est un lieu où cohabitent deux spécialités très différentes : l'informatique et la médecine. Il est donc inévitable que l'incompréhension s'installe parfois entre ces deux domaines. Chacun doit apprendre les limites et impératifs de l'autre lors de la détermination des besoins de chacun. Produire un système complet, qui va fonctionner en situation réelle pour assister un chirurgien par exemple, requiert du temps : il faut comprendre les risques, proposer des outils conformes et adaptés au protocole médical. Qu'attendent les médecins ? Que peut offrir l'outil de RV ? Il est bien souvent difficile de répondre à la question...

## **5.3. Les Appareillages**

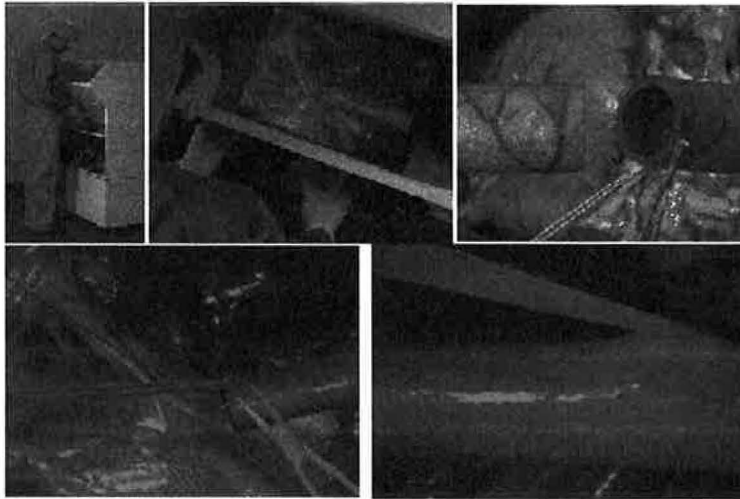


Ces appareils, dérivés du système Phantom, proposent un retour d'effort à l'utilisateur. Ce type de retour autorise la perception de la nature des tissus : élastiques, mous, rigides.

Ces deux appareils sont les plus couramment utilisés pour simuler le travail du chirurgien. Nous allons en découvrir d'autres à travers les applications de la RV dans le monde médical. Pour l'instant aucun d'eux ne simule la pulsation (dans le cas des artères par exemple), la texture des tissus ou encore la température, facteurs pourtant indispensables pour guider chirurgien dans sa zone d'opération.

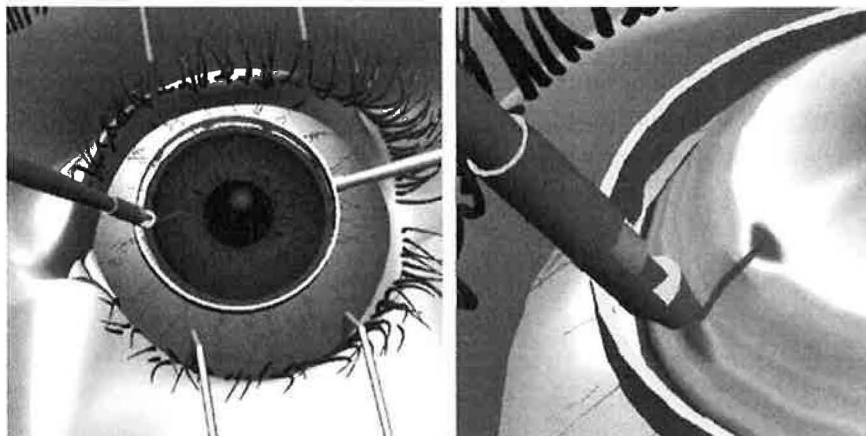
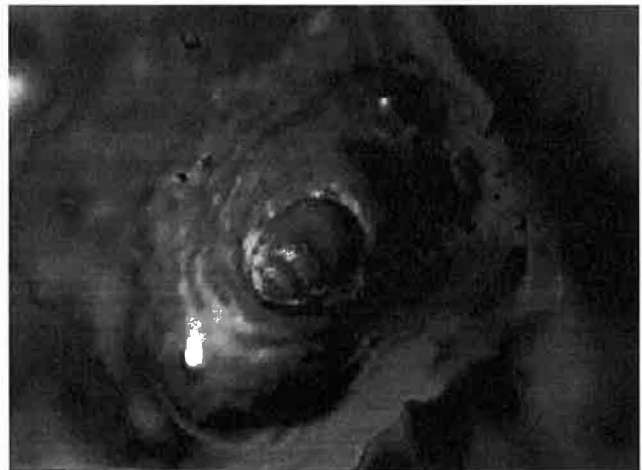


## 5.4. Les Applications



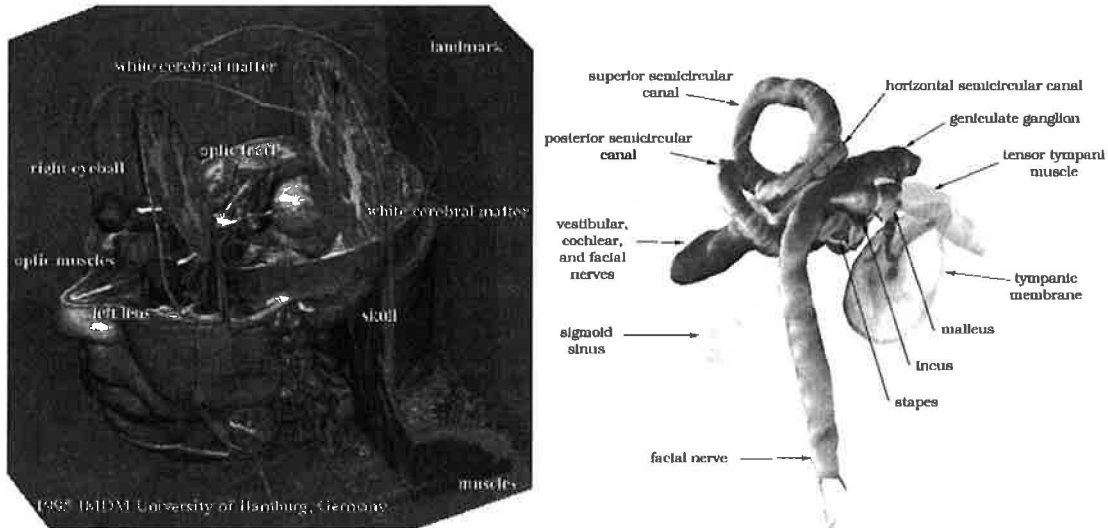
A gauche : chirurgie virtuelle pour suture artérielle (*Boston Dynamics*). L'appareil renvoie la sensation de toucher au travers des instruments. Les tissus virtuels ont un comportement mécanique réel (élasticité, résistance...), le réalisme visuel est obtenu grâce à l'utilisation de textures (photographies de la surface des organes).

L'endoscopie virtuelle permet l'exploration non *invasive* du corps humain. Dans le cas du colon par exemple elle permet de détecter des polypes (foyers potentiels de cancer), qui doivent être détectés le plus tôt possible (les risques augmentent avec la taille). Aucune invasion n'est réalisée, puisque ces simulations sont produites à partir d'examens scanner.



Simulation de chirurgie de l'œil. (Philips Medical System, University of Auckland).

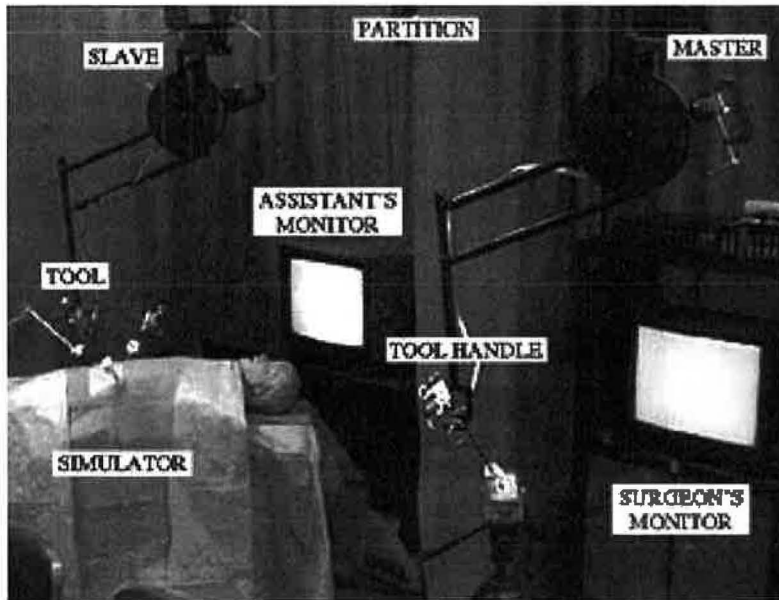
La création d'*atlas* à des fins d'enseignement est également une application importante pour la médecine : des figures annotées et manipulables en trois dimensions sont produites, et peuvent être intégrées dans le processus d'apprentissage des futurs praticiens.



A gauche, une coupe détaillée de la tête. A droite, une représentation annotée de l'oreille interne.

La *Réalité Augmentée* est une forme de réalité virtuelle. Il ne s'agit pas de créer un univers synthétique mais de *superposer* des données à la vision réelle. Dans l'image de droite, on peut apercevoir la tumeur localisée en trois dimensions. Le chirurgien peut ainsi travailler tout en ayant constamment en vue l'objet de l'intervention, et ainsi être guidé de manière plus efficace. Des alarmes peuvent être mises en place pour éviter toute collision avec des tissus périphériques sensibles. (Image courtoisie de Philips Medical System / Brigham and Women's Hospital, Boston). La Réalité Augmentée demande encore beaucoup de recherches, et n'est encore qu'à l'état expérimental.





Tout comme la télérobotique, la téléchirurgie autorisera le contrôle à distance d'un robot qui reproduira les gestes du praticien et qui lui retournera les sensations visuelles et tactiles. Le stimulus visuel peut être renforcé en y associant la stéréoscopie. En ce qui concerne le retour haptique, des améliorations importantes sont encore à venir non seulement pour la gestion de la température ou de la texture mais aussi pour

traiter les problèmes de « décalage » temporel entre l'action du chirurgien et l'action du robot due à la distance. (Image courtoisie du Massachusetts Institute of Technology, programme Human-Machine System Lab Telesurgery). La téléchirurgie n'est encore qu'à l'état expérimental.

## 6. CONCLUSION(S).

Du fait de la disparité des domaines relatifs à la VR que nous avons survolé, il n'y a pas une mais plusieurs conclusions. Il est bien sûr intéressant de discuter du futur de la RV en elle-même, mais il faut également se demander quel impact elle a ou aura sur le monde médical et sur notre société. Il peut être aussi important de préciser que la RV est un concept propre à certaines civilisations ou sociétés, et qu'elle n'est en aucun cas un langage universel comme on l'a longtemps cru pour le film. Nous allons esquisser quelques réflexions sur chacun de ces points afin d'ouvrir un peu plus notre champ de vision à d'autres concepts que ceux purement technologiques.

### 6.1. Le futur de la RV, pour elle-même et pour la médecine.

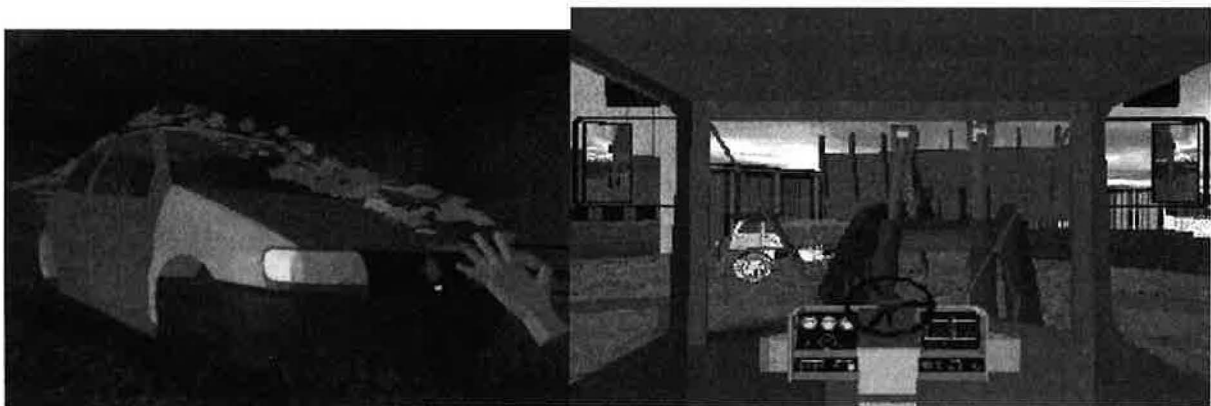
Comme l'a écrit Frederick P. Brooks en 1992 :

*« Although VR has crossed the high pass from 'almost works' to 'barely works', many challenges remain both in the enabling technologies and in the systems engineering and human factors disciplines. »*

« Bien que la RV ait franchi le difficile cap de « fonctionne presque » vers « fonctionne quasiment », plusieurs défis résident encore dans le développement de la technologie, des systèmes et de la connaissance des facteurs humains ».

Frederick P. Brooks

Cette phrase résume encore bien l'état actuel du travail réalisé jusqu'à aujourd'hui. Quoi qu'il en soit, le grand bénéfice de la RV est qu'elle autorise des tests et mises au point sans danger, et avec un minimum d'investissement. La NASA et l'armée sont naturellement très intéressées par ce genre d'application, mais également d'autres entreprises privées : placer des robots dans une usine ou prévoir les accès à divers organes d'un engin motorisé sont autant de points cruciaux à traiter.

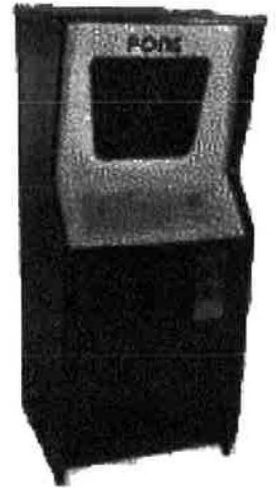


A gauche : test de soufflerie par Volkswagen.

A droite : mise au point de l'ergonomie et de la conception de monte-charge chez Caterpillar.

Pour ce qui est de l'ouverture au grand public aujourd'hui, tout reste encore à faire : « *VR is in the Pong state of video game development* », en référence au premier jeu d'arcade grand public qui sortit en 1972.

Pong : le premier jeu vidéo grand public créé par la firme *Atari* (programmé par Al Alcorn) et sorti en 1972, illustre l'état d'avancement de la Réalité Virtuelle vers le grand public.



Les thèmes d'investigations majeurs restent les suivants en ce qui concerne la technologie des outils d'immersion :

- baisser le temps de réaction du monde virtuel, c'est-à-dire augmenter la puissance de traitement des données en entrée et en sortie ;
- proposer des mondes virtuels « réalistes » ou très détaillés, ce qui revient à développer encore plus les techniques d'accélération graphique ;
- choisir au mieux l'environnement immersif en fonction des applications : CAVE, HMD, bureau virtuel, panorama, ...etc. ;
- améliorer les appareils haptiques pour parfaire les sensations et par là même les réactions des utilisateurs ;
- améliorer les techniques de manipulation et d'accès aux données ;
- mesurer l'illusion d'immersion : mesurer l'impact de chacun des canaux sensoriels peut aider à améliorer notablement l'illusion d'immersion en axant les efforts de recherche ;
- pouvoir mesurer et prédire les éventuels problèmes du mal du simulateur. Il s'agit d'évaluer en détail les effets de l'immersion en phases per et post simulateurs ;
- intégrer des nouvelles technologies comme l'intelligence artificielle, le réseau (travail en groupe ou télé-opération par exemple) pour assister l'expérimentateur et augmenter son efficacité... éventuellement.

L'émergence d'une multitude de pôles de recherche – tout spécialement aux Etats-Unis mais également en Europe – ainsi que de conférences internationales sur la RV montre à quel point la recherche est prolifique sur ce thème. Elle l'est au sens technologique bien sûr mais également au sens de l'écoute entre les concepteurs et les utilisateurs. Sur ce sujet, nous parlons plus particulièrement du monde médical.

Avant tout, et c'est le défi majeur de la RV adapté à l'imagerie médicale, il faut trois choses [45] :

- du réalisme ;
- du réalisme ;
- du réalisme.

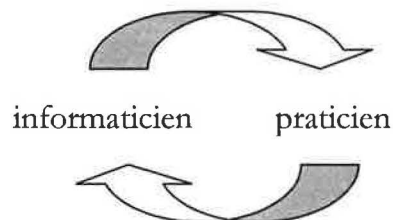
Le praticien – dans le cas du chirurgien par exemple – exige une interaction forte avec le milieu dans lequel il travaille. Les différents tissus doivent se comporter mécaniquement tels qu'ils le feraient dans la réalité : visuellement, mais également haptiquement. L'ergonomie ne doit en aucun cas venir limiter l'efficacité du praticien, et le système ne doit pas alourdir les investissements à réaliser :

- maintenir des coûts faibles pour les simulateurs ;
- rendre les appareils les moins contraignants et les plus intuitifs (ou naturels) possibles.

La considération primordiale est la définition de protocoles médicaux stricts pour valider les techniques d'entraînement par RV et en faire partie intégrante du processus de formation d'un praticien, en d'autres termes :

- valider par l'expérimentation l'efficacité de la formation par RV en phase d'apprentissage ainsi qu'en phase de formation ;
- étudier les mécanismes de perception pour améliorer les retours vers l'individu ; en déduire l'ergonomie des outils d'apprentissage par RV ;
- évaluer les effets physiologiques et psychologiques que peut avoir la technique d'apprentissage et d'entraînement par RV ; comment affectent-ils la formation, les performances individuelles ? ;
- étudier les moyens d'assistance du praticien lors de son intervention.

La boucle de rétroaction suivante est la clé de l'avancée de la RV médicale :



- le praticien doit connaître les limites de l'outil ;
- l'informaticien doit connaître les besoins du praticien ;

Il est difficile de prévoir quel chemin suivra la RV demain : « il n'est pas impossible que d'autres butées technologiques, économiques ou sociologiques ne soient pas encore identifiées, et lorsqu'elles le seront, qu'elles soient infranchissables dans des délais raisonnables. » [1]. Le temps du « *doorway to the future* » où le patient en entrant dans le cabinet du praticien franchira un portail qui l'analysera totalement et immédiatement est encore bien loin : « *20, 50, or perhaps 100 years from now* » [42]. La plupart des techniques de RV médicales n'en sont encore qu'au stade de l'expérimentation, et un long chemin reste à parcourir avant que des protocoles réels ne se mettent en place. En tout cas, au vu des applications actuelles, il semble acquis que la RV jouera un très grand rôle dans la médecine du futur.

## 6.2. Quel impact sur la société ?

Sociologiquement parlant, l'impact de la RV sur la société est à soulever, ne serait-ce que parce qu'elle fait couler beaucoup d'encre dans les media. Cette abondance pourrait tenir en une unique phrase :

« De la même manière ce puissant moyen d'information est un outil d'égale puissance pour la désinformation. » [1]

On ne peut s'empêcher de penser qu'en son temps le cinéma, la radio, la télévision ont été accueillis avec le même scepticisme, la même peur. Ce qui inquiète dans la RV est l'absence de ce que l'on appelle le *proscenium*. Ce mot latin vient lui-même du grec *proskénion*, qui désignait la corniche qui surplombe la scène dans les théâtres antiques. En d'autres termes il n'y a pas de séparation entre les spectateurs et le spectacle. En fait, il n'y a même pas de spectateurs, puisque chacun devient *acteur* et même *réalisateur* puisque il devient possible de choisir sa perspective, son angle de vue et sa position. La RV offre à la fois un spectacle et une expérience du corps.

Au vu de ces considérations, soulignons que l'outil technologique est neutre en soi. Il devient bon ou mauvais selon qui l'utilise. Il n'existe pas de « société régie par les ordinateurs » mais plutôt une société régie par des systèmes mis en place par des hommes. Si l'on songe aux possibilités de la RV dans le futur et à l'influence qu'elle aura sur notre société, il est difficile de pouvoir tout imaginer. La projection des applications potentielles nous semble encore être parfaitement utopique :

- le travail à distance (ou télétravail) qui pourront résoudre les problèmes d'engorgements des voies urbaines, voire modifier la structure familiale puisque l'éloignement ne sera plus requis ;
- un accès plus aisé à l'information, ainsi qu'à l'échange d'idées : des travaux en groupes seront possibles, ainsi que l'*expérimentation* virtuelle. Il existe un proverbe chinois qui illustre la qualité de ce type d'apprentissage :

« L'enseignement qui n'entre que dans les yeux et les oreilles ressemble à un repas pris en rêve »

Preuve, s'il en est, de la nécessité d'avoir d'autres rapports avec la connaissance que la simple imprégnation visuelle ou auditive. Des expériences ont montré que les expériences de RV pleinement immersives s'accompagnaient d'une bien meilleure mémorisation des opérations effectuées.

- des loisirs à part entière : faire de la bicyclette chez soi en parcourant les montagnes, aller au cinéma en s'y immergeant totalement ; préparer une visite touristique ou sa nouvelle maison sont autant de points forts ;
- de nouvelles possibilités artistiques et théâtrales ;
- une communication renforcée puisqu'en cas d'éloignement physique, la RV pourra rendre plus convaincants les réunions, les groupes de travail...etc. ;



Bien que lointains, ces points positifs ont tous leurs pendants négatifs, mais c'est à nous d'en décider. Les visions pessimistes ne manquent pas sur le sujet... Les jeux vidéos par exemple recherchent l'immersion la meilleure possible – dans une limite primordiale de coût –, est-ce bon ? est-ce mauvais ? C'est le dosage qui est la clé du succès : il est très gratifiant de pouvoir ressentir plus que des effets visuels dans un loisir, car nous y découvrons alors d'autres dimensions :

« Un espace à dimensions multiples n'est pas plus riche. Il est autre qu'un espace à deux ou trois dimensions. Il ne s'agit pas d'addition de qualités. Dans un espace pourvu de X dimensions, toutes les dimensions sont modifiées. » [2]

Et donc le plaisir que l'on peut en retirer, ce qui, au passage, est typique d'une civilisation certes technologiquement avancée mais surtout peut-être d'une civilisation *occidentalisée*. Si ce loisir devient une drogue ou qu'il devient un échappatoire de la réalité, il devient alors mauvais.

Le jugement final nous appartient, ou plutôt il faut nous en donner les moyens. C'est par notre volonté seule que nous devons accepter l'immersion que nous propose la RV, et non pas par la passivité. C'est ce qui nous amène enfin à parler du concept de « *willing suspension of disbelief* ».

### **6.3. La RV, une nouvelle réalité oui mais... le « *willing suspension of disbelief* ».**

C'est un poète anglais du XIX<sup>ème</sup> siècle – Samuel Taylor Coleridge – qui est à l'origine de cette expression. Nous pourrions la traduire par « *la volonté d'y croire* ». Par cette phrase S. Taylor désigne le processus par lequel l'imagination du lecteur l'entraîne dans le poème, mais au sens où ce dernier accepte consciemment de se laisser transporter par la poésie. En d'autres termes c'est lui qui décide de se plonger dans l'histoire pour en ressentir le plus de plaisir. Un article de *Digital Culture* du 17 mai 1995 relate la récupération de ce concept par les fabricants de jeux vidéo – Sony, Sega et 3DO en l'occurrence. Le PDG de Sega, Tom Kalinske, disait en substance :

« *Consumers will demand immersive experiences that create suspension of disbelief.* »

L'omission du mot *willing* qui suppose la pleine volonté du participant était on ne peut plus remarquable. Néanmoins il était rapporté que les jeunes joueurs (10-15 ans) connaissaient les limites de ces notions, et qu'ils n'étaient pas aussi passifs que ce à quoi l'on pourrait s'attendre. Une telle attitude doit nous rappeler que les débordements dont nous entendons parler ne sont que le résultat d'actes de personnes mentalement fragiles ou vivant dans un contexte social marginal.



Pour en revenir à l'expression de S. Taylor, la RV actuelle est encore bien loin de pouvoir tromper nos sens. Les appareils restent contraignants à utiliser, la qualité visuelle est perfectible, les sens haptiques sont encore très peu exploités. Comme il a été dit auparavant, la RV n'est pas un *medium* universel. Il faut d'abord une période d'apprentissage, une phase où l'on doit distinguer la forme du fond. L'utilisateur devra faire un effort conscient pour accepter la réalité du monde synthétique, en comprenant les limitations et les moyens de déplacement qui sont encore peu intuitifs. Lorsque nous allons au cinéma, nous avons l'impression que ce que nous voyons est un transfert de la réalité parce que nous comprenons l'outil qui est derrière. Nous l'acceptons à un point tel que nous analysons directement le contenu des images en les ramenant à une autre « réalité », celle que nous imaginons. Des expériences ont été menées à la Sorbonne sur des civilisations peu avancées en technologie : des « primitifs » ont été soumis à une projection de film dont ils n'ont pu comprendre le contenu, tant les conditions de vision leur était peu familière [2]. Ce phénomène s'est rencontré pour la photographie également, où l'on rapporte que des habitants de Nouvelle Guinée ne se reconnaissaient pas sur un Polaroid.

« Tout signe, pour être lu, exige un effort de reconnaissance. » [2]

La RV « universelle » sera celle qui s'affranchira de la prise en compte de la forme. Nous en sommes encore bien loin. C'est pourtant le cas de l'*holodeck*, bien connu des habitués de *Star Trek*, la célèbre série américaine de science-fiction, qui immerge totalement les personnages dans un univers virtuel. Nous en livrons ici quelques images, qui seront peut-être la réalité d'après-demain ...



Dans *Star Trek : Generations*, l'équipage de l'*USS Enterprise* s'autorise un moment de détente (images du haut). Le capitaine Picard quitte la simulation (images du bas). Cette fiction nous présente une réalité virtuelle totalement immersive, où l'on s'est totalement affranchi de la forme : nulle connaissance d'aucun outil d'immersion n'est requis, et aucune contrainte extérieure n'est imposée aux utilisateurs.

## 7. GLOSSAIRE.

### A

**Accélérateur** : Matériel spécifique chargé d'aider l'ordinateur à calculer plus rapidement, par exemple pour l'affichage tridimensionnel.

**Accommodation** : Changement de focale de l'œil par modification de la forme du cristallin afin de maintenir une vue nette sur un objet proche.

**Actuateur** : Système hydraulique ou électrique chargé du retour de force.

**Affichage à Cristaux Liquides (LCD : Liquid Cristal Display)**: Appareil d'affichage qui emploie un film bipolaire pris en sandwich entre deux plaques de verre. Il est léger, transmissif ou réfléchissant et est souvent employé dans les HMD.

**Affichage Monté sur Tête (HMD : Head Mounted Display)** : Lunettes ou casque dotés d'un écran par œil, qui permet de voir le monde virtuel. Un tel appareil est souvent couplé à un capteur de position pour suivre les mouvements de la tête.

**Affichage Tactile** : Appareil fournissant à la fois des sensations tactiles et kinesthésiques.

**Affichage Tête Haute (HUD : Head Up Display)** : Système d'affichage superposant des données graphiques à la vue du monde réel. (voir *Réalité Augmentée*).

**Agent Assistant** : Algorithme d'intelligence artificielle chargé d'assister ou de guider l'utilisateur dans le monde virtuel.

**Autonomie** : Action et comportement par lui-même d'un objet virtuel (sans dépendance vis-à-vis de l'utilisateur).

**Avatar**: Objet virtuel qui personnifie l'utilisateur dans le monde synthétique.

### B

**Baguette Magique** : Appareil tridimensionnel utilisé pour pointer et interagir avec le monde virtuel.

**Binoculaire** : Affichage d'images différentes pour chaque œil en vue de produire un effet stéréoscopique.

**Bi-oculaire** : Affichage de la même image pour l'œil gauche et l'œil droit. Souvent utilisé lorsque la stéréoscopie n'est pas requise.

**BOOM (Binocular Omni-Orientation Monitor)** : Appareil d'affichage tridimensionnel monté sur bras hydraulique. La position même du bras supportant le dispositif de vision renseigne l'ordinateur sur la position réelle de l'utilisateur.

## C

**Capteur de position** : Appareil fournissant une information de position et d'orientation.

**CAVE** : Monde Virtuel projeté sur les murs, plafonds et sols d'une pièce pour donner l'illusion d'immersion.

**Champ de vision** : Donné par l'angle, en degrés, de la zone couverte par le regard. L'être humain dispose d'un champ de vision de  $180^\circ$  environ, dont  $140^\circ$  par œil avec une zone de recouvrement. La sensation d'immersion apparaît lorsque le champ de vision est supérieur à environ  $70^\circ$ .

**Chirurgie Minimale Invasive Virtuelle (Virtual MIS : Virtual Minimal Invasive Surgery)** : Utilisation de la RV pour proposer virtuellement des procédures chirurgicales limitant l'accès au corps du patient.

**Conflit Kinesthésique** : Absence de corrélation entre le toucher/mouvement virtuel et le toucher/mouvement réel.

**Conflit Sensoriel** : Théorie selon laquelle le mal du simulateur est causé par le fait que le corps essaie d'interpréter des données conflictuelles fournies par les sens.

**Convergence** : Se produit en vision stéréoscopique lorsque l'image perçue par les yeux se fond en une seule

**Convolvotron** : Système contrôlant le son dans un système de RV.

**Coordonnées** : Données qui déterminent la position des objets dans le monde virtuel. Le nombre de ces coordonnées détermine la *dimension* de l'espace.

**Cortex Occipital** : Situé derrière le cerveau, il est le centre de traitement de la projection rétinotopique.

**Cortex Pariétal** : Zone du cerveau voisine du cortex occipital qui traite les informations de position et de direction.

**Cyberespace** : Monde synthétique.



## D

**Distorsion métallique** : Interférences ou dégradation de performance des capteurs de mouvement magnétiques dues à la présence d'objets métalliques.

**Dynamique** : Règles qui gouvernent toute action et comportement dans le monde synthétique.

## E

**Eclairage dynamique** : Changements dans les effets de lumière de manière interactive lorsque les objets sont déplacés.

**Effecteurs** : Appareils utilisés pour les entrées/sorties, comme les gants, les détecteurs de position, les écouteurs, les écrans...etc.

**Effet Doppler** : Augmentation ou diminution apparente de la fréquence d'un effet sonore ou lumineux à mesure que celui-ci se rapproche ou s'éloigne de l'observateur.

**Environnement** : Modèle généré par ordinateur qui constitue le monde dans lequel évolue l'utilisateur.

**Environnements Virtuels** : Simulation Réaliste de scènes interactives.

**Exosquelette** : Structure mécanique utilisée pour le contrôle ou le retour d'effort.

## G

**Gant Virtuel** : Gant doté de capteurs et qui permet à l'utilisateur d'interagir avec le monde virtuel.

**Graphisme Tridimensionnel** : Représentation de la géométrie sur une surface plane de telle manière qu'elle semble être en trois dimensions.

## I

**Immersion**: Sensation de faire partie du monde virtuel.

**Interface** : Ensemble de logiciels, ordinateur et appareils qui permettent le dialogue entre l'homme et la machine.

**Interface Haptique** : Utilisation de capteurs de toucher (peau) et d'appareils à retour d'efforts (muscles, articulations).

## K

**Kinesthésie** : Sensations produites par muscles, tendons et articulations qui sont stimulées par le mouvement.

## L

**Latence** : Temps de retard entre le mouvement réel de l'utilisateur et sa prise en compte par le gestionnaire de mouvement.

**Lobe Temporal** : Zone du cerveau placée devant le cortex occipital et le cortex pariétal chargé de l'audition.

**Lumière Ambiante** : Illumination provenant de l'extérieur (du monde réel).

**Lunettes à Obturateurs** : Lunettes masquant alternativement l'œil gauche et l'œil droit en synchronisation avec l'affichage qui produit deux images légèrement différentes. Elles permettent la stéréoscopie.

## M

**Maladie du Simulateur** : Effets secondaires allant d'un léger malaise jusqu'à de fortes nausées, pouvant être causés par différents facteurs liés à la simulation.

**Manette de Jeu / Joystick** : Périphérique d'entrée comportant un levier qui permet de se diriger dans la direction souhaitée, en n'utilisant qu'une seule main.

**Modèle** : Simulation par l'ordinateur d'un objet ou phénomène réel.

**Mondes Partagés** : Environnements virtuels pouvant accueillir plusieurs participants.

**Moteur de Réalité (Reality Engine)** : Système informatique chargé de générer les objets virtuels, habituellement en temps réel.

## N

**Navigation** : Déplacement à travers le monde virtuel.

**Niveau de Détail (LOD : Level Of Detail)** : Technique de simplification d'objets virtuels tridimensionnels qui régule la qualité en fonction de la distance de vision. Permet d'accélérer le traitement d'affichage.

## O

**Objet** : Forme virtuelle qui peut être manipulé dans le monde synthétique.

**Occlusion**: Masquage partiel ou complet d'un élément.

**Ombre de Gouraud** : Technique d'éclairage d'un objet virtuel pour améliorer la sensation de relief.

**Ombre de Phong** : Autre méthode de simulation d'éclairage sur les objets pour donner la sensation de profondeur. (voir *Ombre de Gouraud*).

## P

**Parallaxe** : Différence de vue de chaque œil du fait de leur position, qui crée l'impression de profondeur.

**Parallaxe Binoculaire** : Voir parallaxe et parallaxe de mouvement.

**Parallaxe de Mouvement** : Moyen qu'a l'œil pour évaluer les distances, en remarquant que les objets proches semblent se déplacer plus vite que les objets lointains.

**Perception de profondeur** : Techniques qu'emploie le cerveau pour avoir une idée de la position spatiale d'un objet : les ombres, les reflets, la position relative ...etc.

**Perspective**: Règles déterminant la taille relative des objets (ainsi que leur distorsion) sur une surface plane pour donner l'illusion de profondeur.

**Photoréalisme** : Tentative de production d'images réalistes, avec beaucoup de détails et de textures.

**Pixel (Picture Element)** : Le plus petit élément d'affichage.

**Placage de Texture** : Placage d'une image de matériau (photo) sur un objet pour le rendre plus réaliste.

**Plate-forme** : Système mécanique fournissant un mouvement réel qui vient améliorer la sensation obtenue visuellement.

**Polygone** : Surface délimitée par une ligne brisée fermée et servant d'élément de base (souvent le triangle) pour les objets virtuels.

**Portail** : Zone du monde virtuel permettant de changer de monde ou d'enclencher un action.

**Présence**: Sensation d'être immergé dans un environnement.

## R

**Rapport d'aspect** : Rapport entre la hauteur et la largeur du champ de vision.

**Réalité Artificielle** : voir *Réalité Virtuelle*.

**Réalité Augmentée** : Utilisation de lunettes transparentes avec en surimpression des informations données par l'utilisateur. (voir aussi *affichage tête haute*).

**Réalité Virtuelle (RV)** : Système informatique chargé de gérer un monde virtuel dans lequel est plongé un utilisateur de manière pleinement interactive.

**Résolution** : Nombre de pixels d'un afficheur ou d'une image.

**Retour de force/d'effort** : Pression, vibration ou force renvoyée par un appareil pour augmenter la sensation de réalisme. On distingue le *retour d'effort* du *retour tactile* en ce sens que ce dernier ne concerne que les sensations cutanées.

## S

**Six Degrés de Liberté (6DOF : 6 Degrees of Freedom)** : Possibilité de se déplacer suivant trois directions spatiales et selon trois axes d'orientation.

**Spatialisation** : Traitement sonore pour donner l'illusion de sources localisées dans l'espace.

**Suivi Tête** : Technique de suivi de positionnement et d'orientation de la tête.



## T

**Taux de Rafraîchissement** : Fréquence à laquelle une image est renouvelée sur une surface d'affichage.

**Télé-manipulation** : Contrôle à distance (avec un robot par exemple).

**Télé-opérateur** : Personne réalisant la télé-manipulation.

**Téléprésence**: Télé-manipulation couplée à un retour sensoriel de l'environnement distant.

**Temps Réel** : Action qui s'exécute sans qu'aucun retard ne soit perceptible entre la requête et la réponse.

## U

**Univers** : Ensemble composé du monde virtuel et de tout ce qu'il renferme.

## V

**Vection**: Sensation de mouvement causée par le seul stimulus visuel.

## **8. QUELQUES ADRESSES INTERNET POUR ALLER PLUS LOIN.**

Le matériel, la recherche et les possibilités de la RV :

<http://www.shef.ac.uk/uni/projects/vrmbg/index.html>  
<http://www.iitvr.com/inputs.html>  
[http://www.athomas.demon.co.uk/vr/VR\\_Interface.html](http://www.athomas.demon.co.uk/vr/VR_Interface.html)  
[http://www.simteam.com/autourVR/CoursVR/Intro/vrintro\\_2.htm](http://www.simteam.com/autourVR/CoursVR/Intro/vrintro_2.htm)  
<http://www.5dt.com/products.html>  
<http://www.vruniverse.com/vrlinks.html>

Les applications médicales :

<http://www.smd.be/msw/fra/r&d/telemed/telf.htm>  
[http://www.nlm.nih.gov/research/visible/visible\\_human.html](http://www.nlm.nih.gov/research/visible/visible_human.html)  
<http://www.man.ac.uk/MVC/research/>  
[http://www.hitl.washington.edu/projects/knowledge\\_base/medapps.html](http://www.hitl.washington.edu/projects/knowledge_base/medapps.html)  
[http://home.planet.nl/~frans.gerritsen/DSP\\_991109/index.htm](http://home.planet.nl/~frans.gerritsen/DSP_991109/index.htm)  
<http://www.informatik.umu.se/~jwworth/medpage>

L'histoire de l'informatique :

<http://histoire.info.online.fr/prehistoire.html>

## 9. BIBLIOGRAPHIE.

- [1] G. Burdea, P. Coiffet, « La Réalité Virtuelle », Hermès, Paris, 1993
- [2] G. Francastel, « L'Image, La Vision et L'Imagination : De la Peinture Au Cinéma », Denöel/Gonthier, 1983
- [3] M. Göbel, « Projects in VR », IEEE Computer Graphics and Applications, pp. 10-13, Volume 16, Number 1, January 1996
- [4] P. Dev, « Imaging and Visualization in Medical Education », IEEE Computer Graphics and Applications, pp.21-31, Volume 19, Number 3, May/June 1999
- [5] J. Charles, « Virtually There : Taking VR to Work », IEEE Computer Graphics and Applications, pp.4-9, Volume 17, Number 6, November/December 1997
- [6] R. Baldwin, « VR : Friend or Foe ? », IEEE Computer Graphics and Applications, pp.102-103, Volume 17, Number 6, November/December 1997
- [7] Y. Ikei, K. Wakamatsu and S. Fukuda, « Vibratory Tactile Display of Image-Based Textures », IEEE Computer Graphics and Applications, pp.53-61, Volume 17, Number 6, November/December 1997
- [8] S. L. Dawson, J. A. Kaufman, « The Imperative for Medical Simulation », IEEE Special Issue on Virtual & Augmented Reality in Medicine, pp. 479-483, Volume 86, Numero 3, March 1998
- [9] R. M. Satava, S. B. Jones, « Current and Future Applications of Virtual Reality for Medicine », pp. 484-489, Volume 86, Numero 3, March 1998
- [10] E. Chen, B. Marcus, « Force Feedback for Surgical Simulation », pp. 524-529, Volume 86, Numero 3, March 1998
- [11] Z. Soferman, D. Blythe, « Advanced Graphics Behind Medical Virtual Reality : Evolution of Algorithms, Hardware, and Software Interfaces », pp. 531-554, Volume 86, Numero 3, March 1998
- [12] J. W. Hill, « Telepresence Technology in Medicine : Principles and Applications », pp. 569-580, Volume 86, Numero 3, March 1998
- [13] N. Ezquerra, I. Navazo, T. I. Morris and E. Monclus, « Graphics, Vision, and Visualization in Medical Imaging : A State of the Art Report », pp.21-80, Eurographics'99 STAR, 20<sup>th</sup> Annual Conference of The European Association for Computer Graphics « Bringing to a New Life our Cultural Heritage », 7—11 September 1999, Milano
- [14] H. Iwata, « The Torus Treadmill : Realizing Locomotion in Ves », IEEE Computer Graphics and Applications, pp. 30-35, November/December 1999
- [15] H. Hoffman, J. Groen, S. Rousseau, A. Hollander, W. Winn, M. Wells and T. Furness, « Tactile Augmentation : Enhancing Presence in Virtual Reality with Tactile Feedback from Real Objects" » Journal of American Psychological Society, 1996, San Francisco, CA
- [16] S. Rosenberg, « The New Video-Game Philosophers : Did Someone Say Suspension of Disbelief ? », Digital Culture, May 17, 1995
- [17] F. P. Brooks, Jr., « What's Real About Virtual Reality ? », IEEE Computer Graphics and Applications, pp. 16-27, November/December 1999
- [18] I. E. Sutherland, « The Ultimate Display », Information Processing 1965 : Proceedings of IFIP Congress 65, Volume 2

- [19] E. M. Kolasinski, S. L. Goldberg, J. H. Hiller, « Simulator Sickness in Virtual Environments », U.S. Army Research Institute for the Behavioral and Social Sciences, Technical Report 1027, May 1995
- [20] G. Burdea, « Virtual Reality Systems and Applications », Electro'93 International Conference, Short Course, Edison NJ, 28<sup>th</sup> April, 1993
- [21] F. Brooks *et al.*, « Project GROPE – Haptic Displays for Scientific Visualization », Computer Graphics, Volume 24, Number 4, pp. 177-185
- [22] B-T Phong, « Illumination for Computer Generated Pictures », CACM, 18(6), pp. 311-317, June 1975
- [23] H. Gouraud, « Continuous Shading of Curved Surfaces », IEEE Transactions on Computers, C-20(6), pp.623-629, June 1971
- [24] W. Gibson, « Neuromancien », Edition J'ai Lu
- [25] J. G. Casali, « Vehicular Simulation-Induced Sickness, Volume 1 : An Overview. », IEOR Technical Report 8501 (Naval Training System Center TR 86-010), August 1986
- [26] R. S. Kennedy, J. E. Fowlkes, « Simulator Sickness is Polygenic and Polysymptomatic : Implications for Research », International Journal of Aviation Psychology, 2(1), pp. 23-38, 1992
- [27] R. S. Kennedy, J. E. Fowlkes, K. S. Berbaum and M. G. Lilienthal, « Use of a Motion Sickness History Questionnaire for Prediction of Simulator Sickness », Aviation, Space and Environmental Medicine, 63(7), pp. 588-593, 1992
- [28] M. E. Mc Cauley, T. J. Sharkey, « Cybersickness : Perception of Self-Motion in Virtual Environments », Presence, 1(3), pp. 311-318, 1992
- [29] M. Treisman, « Motion Sickness : An Evolutionary Hypothesis », Science, 197, pp. 493-495, 1977
- [30] T. A. Stoffregen, G. E. Riccio, « An Ecological Theory of Orientation and the Vestibular System" » Psychological Review, 95(1), pp. 3-14, 1988
- [31] T. A. Stoffregen, G. E. Riccio, « An Ecological Critique of the Sensory Conflict Theory of Motion Sickness" » Ecological Psychology, 3(3), pp. 15-194, 1991
- [32] J. T. Reason, J. J. Brand, « Motion Sickness », London : Academic Press, 1975
- [33] E. C. Regan, « Side-Effects of Immersive Virtual Reality », International Applied Military Psychology Symposium, July 26-29, 1993
- [34] R. M. Stern, S. Hu, R. Leblanc and K. L. Koch, « Chinese Hyper-Susceptibility to Vection-Induced Motion Sickness », Aviation, Space and Environmental Medicine, (9), pp. 827-830, 1993
- [35] R. S. Kennedy, K. S. Berbaum, M. G. Lilienthal, W.P. Dunlap, B. F. Mulligan and J. F. Funaro, « Guidelines for Alleviation of Simulator Sickness Symptomatology », (NTSC TR 87-007), March, 1987
- [36] K. C. Uliano, E. Y. Lambert, R. S. Kennedy and D. J. Sheppard, « The Effects of Asynchronous Visual Delays on Simulator Flight Performance and the Development of Simulator Sickness Symptomatology », (NTSC 85-D-0026-1), 1986
- [37] R. S. Kennedy, L. H. Franck, « A Review of Motion Sickness with Special Reference to Simulator Sickness », Workshop on Simulator Sickness, 26-28 September, 1983

- [38] D. E. Parker, D. L. Harm, « Mental Rotation : A Key to Mitigation of Motion Sickness in the Virtual Environment », *Presence*, 1(3), pp. 329-333, 1992
- [39] J. Ehrlich, M. J. Singer and S. Cinq-Mars, « Stereoscopic and Monoscopic Displays in Virtual Environments », ARI Technical Report, Research Institute for the Behavioral and Social Sciences, Alexandria, VA
- [40] K. R. Boff, J. E. Lincoln, « Engineering Data Compendium : Human Perception and Performance », AAMRL, Wright-Patterson AFB, OH, pp. 166-191, 1988
- [41] T. J. Sharkey, M. E. McCauley, « Does a Motion Base Prevent Simulator Sickness ? », *Proceedings of the AIAA/AHS Flight Simulation Technologies Conference*, Report AIAA-92-4133-CP, Washington D.C., pp. 21-28, 1992
- [42] R. M. Satava, « Virtual Reality Surgical Simulator – The First Steps », *Proceedings of VR Systems '93 Conference*, New York, pp. 41-49, March, 1993
- [43] Y. Ikei, K. Wakamatsu and S. Fukuda, « Vibratory Tactile Display of Image-Based Textures », *IEEE Computer Graphics and Applications*, pp. 53-61, Volume 17, Number 6, November/December 1997
- [44] I. Asimov, « *Fantastic Voyage* », Boston, MA, Houghton-Mifflin, 1966
- [45] W. M. Smith, « Scanning the Technology : Engineering and Medical Science Chart *Fantastic Voyage* », *IEEE Special Issue on Virtual and Augmented Reality in Medicine*, pp.474-478, Volume 86, Number 3, March 1998

## **2. Tour d'Horizon de la Représentation Tridimensionnelle.**

*A triangle was an improvement to the square wheel.  
It eliminated one bump*

## 1. INTRODUCTION.

La manipulation des entités tridimensionnelles est largement explorée, tant son rôle est prépondérant dans l'appréhension de l'information : la géographie, la physique, la chimie, l'aéronautique, la médecine ne sont que quelques exemples des champs scientifiques concernés. Ce large spectre rend d'autant plus aiguës les problématiques de performance, de coût mémoire ainsi que d'adéquation de représentation. La visualisation scientifique met au défi les capacités algorithmiques et matérielles sur plusieurs fronts :

- la taille des données augmente du fait notamment de la capacité accrue des machines à traiter l'information mais aussi des progrès en numérisation ;
- certaines applications requièrent un niveau de détail important afin d'améliorer précision et réalisme ;
- l'observateur doit interagir avec l'information en temps réel afin d'en comprendre tous les aspects et d'en extraire tout le potentiel.

Nous discuterons dans ce chapitre des différentes techniques utilisées pour acquérir, représenter et afficher les objets tridimensionnels sur un ordinateur. En effet, la chaîne conduisant d'un objet réel ou synthétisé à son affichage est la suivante :

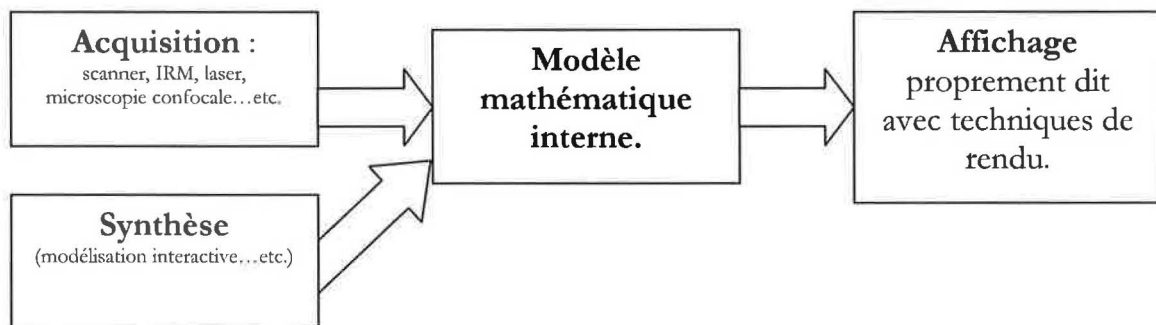


Figure 1

Nous diviserons ce tour d'horizon en deux parties : la première traitera de la représentation *volumique*, la seconde de la représentation *surfactive*, dans un contexte uniquement tridimensionnel.

Le thème de rendu ne sera pas développé puisqu'il constitue une problématique trop large pour le cadre de notre approche.

## 2. REPRÉSENTATION VOLUMIQUE.

Nous étudierons dans cette section les modèles de description et de projection qui tiennent compte de l'intérieur de l'objet, offrant ainsi une représentation très riche au détriment d'un coût en espace mémoire important.

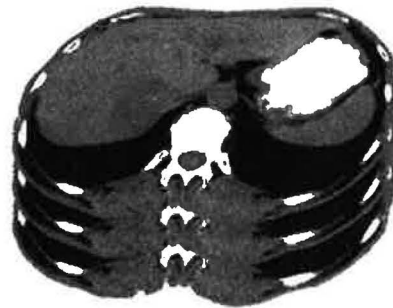
### 2.1. Acquisition volumique des données

La phase d'acquisition impose la nature, la quantité et la précision des données de volume. De ce fait les modalités de numérisation méritent que l'on s'y attarde quelque peu car elles mettent en place les contraintes qui guideront les algorithmes de reconstruction du volume.



Le *microscope confocal* permet l'acquisition de « tranches » d'objets en prenant des photographies d'altitudes des objets observés : l'empilement de ces images fournit le relief de la surface. L'image suivante montre une reconstruction tridimensionnelle de cellules d'*Arabidopsis* (Image courtoisie de *University of Cambridge, Department of Plant Sciences, Jim Haseloff*).

Le *scanner médical* (à rayons X) ou l'IRM (Imagerie par Résonance Magnétique) numérise véritablement le volume puisqu'il fournit des sections du corps humain après traitement mathématique. Chaque point renseigne sur les propriétés locales des tissus observés. Contrairement à l'acquisition par microscopie confocale, une information supplémentaire est donc à disposition, qui permet ensuite la reconnaissance des tissus par des techniques automatiques de segmentation.





## 2.2. Modèles internes de représentation volumique.

### 2.2.1. Les voxels.

Ils désignent des éléments de volume au sein desquels les propriétés restent constantes (couleur, densité, position...etc.). C'est un modèle direct puisqu'il stocke explicitement chaque point de l'objet, y compris son intérieur. Les informations rattachées à chaque voxel sont très variables car spécifiques à l'utilisation de ces données.



A gauche : vue d'un foie tumoral sous forme de voxels. Des propriétés de couleur et de transparence ont été ajoutées à l'information spatiale. L'image met en évidence l'aspect discret d'une telle technique : il est facile de distinguer les petits carrés ainsi que les « marches d'escalier » sur cette vue.

Il est bien sûr possible de limiter ces effets indésirables, mais c'est au prix d'une augmentation du nombre de voxels.

### 2.2.2. Les surfaces implicites.

Les *fonctions potentiels* sont des outils pratiques pour le modelage de solides en ce sens qu'elles définissent à la fois la surface et l'intérieur de l'objet (et par la même son extérieur).

Elles s'emploient principalement pour modéliser les objets organiques, présentant peu de discontinuités. Le modelage s'apparente au travail de la pâte à modeler ou de liquide très visqueux. C'est Blinn [32] qui introduisit les surfaces implicites en informatique graphique, idée reprise et développée [33], [34] ensuite pour aboutir à un modèle géométrique résultant de l'emploi de plusieurs éléments de base (sphères, ellipsoïdes, tubes... etc.) décrits de manière implicite.

Par exemple, l'équation  $f(x,y,z) = 0$ , où  $f: \mathbb{R}^3 \rightarrow \mathbb{R}$  est la fonction suivante :

$$f(x,y,z) = (y-1)^2 + (z+2)^2 - 7$$

décrit la surface d'un cylindre parallèle à l'axe des  $x$ , comme montré dans la figure ci-dessous :

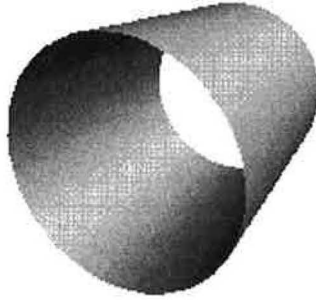


Figure 2

La surface du cylindre est en fait définie par l'ensemble des points vérifiant la relation :

$$\{(x, y, z) \in R^3 \mid f(x, y, z) = 0\}$$

Contrairement à une définition *explicite* ou *paramétrique*, la surface est ici décrite par une *relation algébrique* entre les variables.

L'intérêt des surfaces implicites réside dans le fait que si deux formes sont définies par les relations  $f(x, y, z) = 0$  et  $g(x, y, z) = 0$ , alors le solide défini par  $f(x, y, z) + g(x, y, z) = 0$  est le *mélange* des deux formes initiales. Cette opération est baptisée *union par mélange*. Par exemple, considérons la fonction suivante :

$$D(r) = 1/r^2$$

Où  $r$  est la distance entre un point  $(x, y, z)$  de l'espace  $R^3$  et un point  $C$  particulier. Si l'on recherche les zones de l'espace où  $D(r) = 1$  par exemple, nous obtenons :



Figure 3

Qui définit bien une sphère si  $r$  est la distance euclidienne classique entre  $P$  et  $C$ .  
 A présent, si l'on place deux points  $C_1$  et  $C_2$ , voici ce qui se produit pour la surface définie par  $D(r_1) + D(r_2) = 1$  :

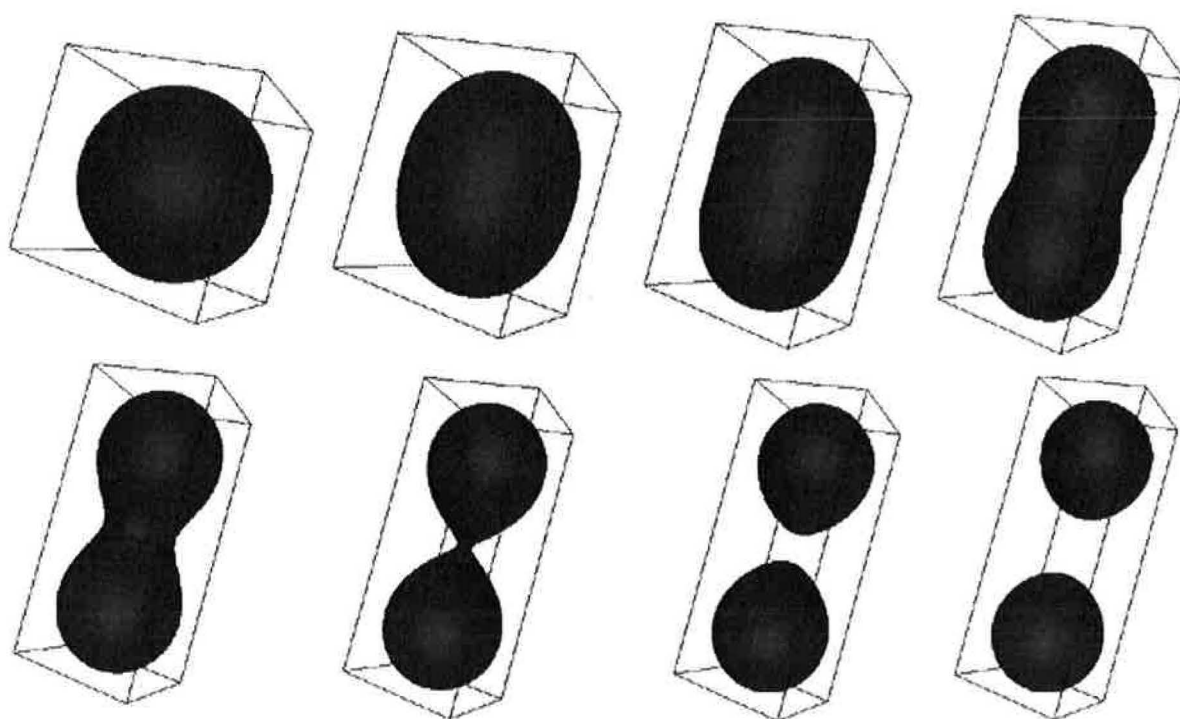


Figure 4

La figure nous présente plusieurs possibilités, selon que  $C_1$  et  $C_2$  sont plus ou moins proches dans l'espace. Il se crée alors des zones d'influences, et la valeur au point  $P$  donné – encore nommée *potentiel* – est la somme des deux valeurs de distance entre chaque centre  $C_i$ . Blinn nomma ce modèle *Blinn Objects* ou *Blobs*. On trouve également les terminologies *metaballs* ou *soft objects*.

La classe des *surfaces implicites à potentiel échantillonné* [36][37] autorise une évaluation rapide et donc un rendu accéléré de la surface implicite en interpolant la surface entre les valeurs stockées aux sommets des cellules d'une grille 3D. Cette méthode dispose de plus d'améliorations importantes limitant l'évaluation des cellules et le coût en mémoire du stockage des potentiels [38].

Il devient alors possible, en variant les fonctions de distances, de combiner différentes équations – appelées alors *primitives* – afin de créer des objets plus complexes comme pour ce visage très connu par exemple [34], dû à S. Muraki :

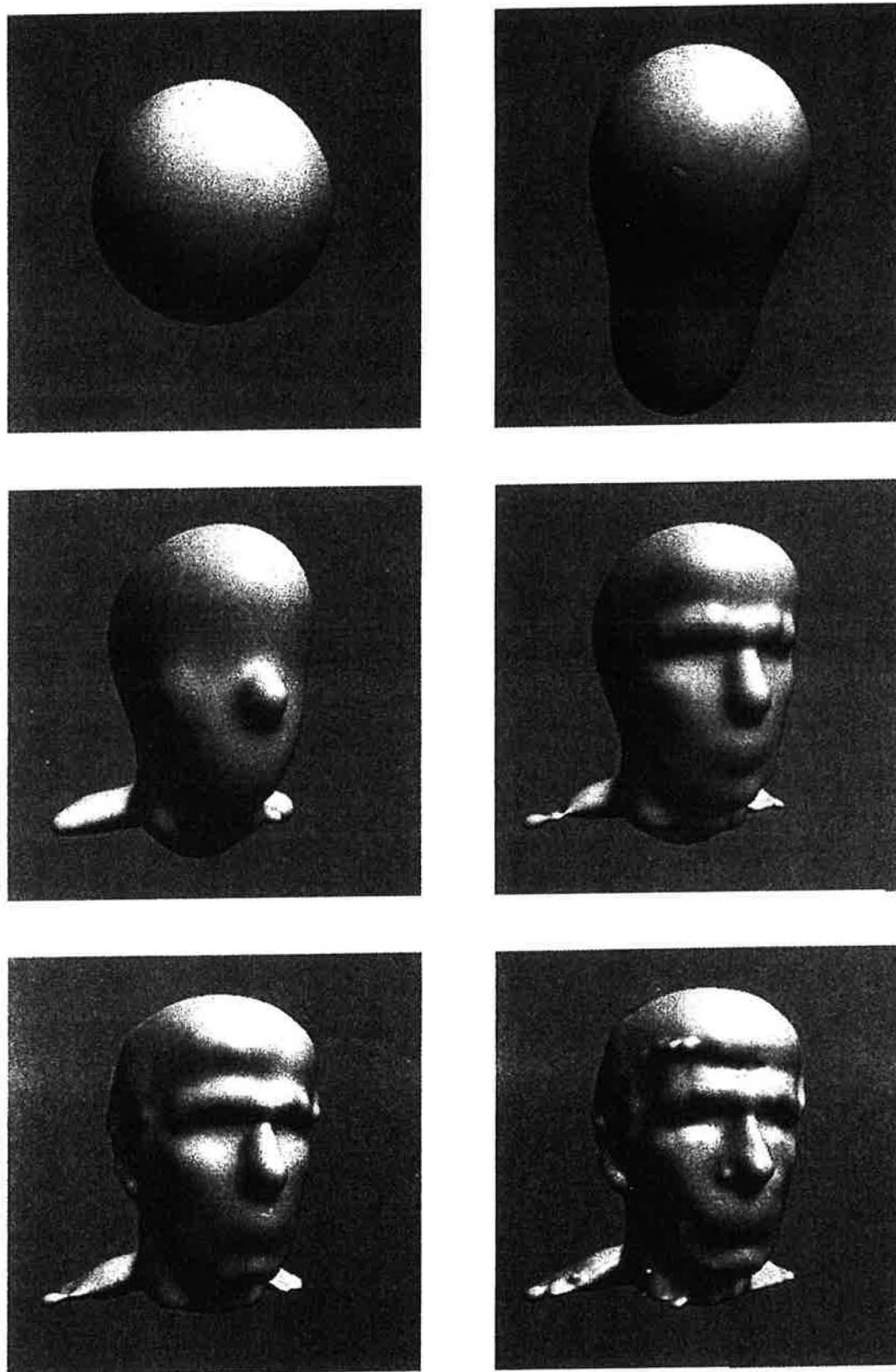


Figure 5

Chaque primitive peut être déformée selon les besoins de reconstruction, comme nous le verrons dans le chapitre traitant de l'ajustement surfacique.

Avantages :

- déterminer l'appartenance d'un point à la surface (ou l'intérieur) d'un objet est immédiat ;
- l'intersection de deux surfaces d'objets est déterminée efficacement.

Inconvénients :

- certaines formes complexes sont difficiles à décrire avec ce formalisme (notamment les angles aigus par exemple) ;
- les propriétés dépendent du système de coordonnées ;
- les objets complexes sont parfois lourds à évaluer (évaluation d'une fonction par primitive présente).

### **2.3. Projection volumique des données.**

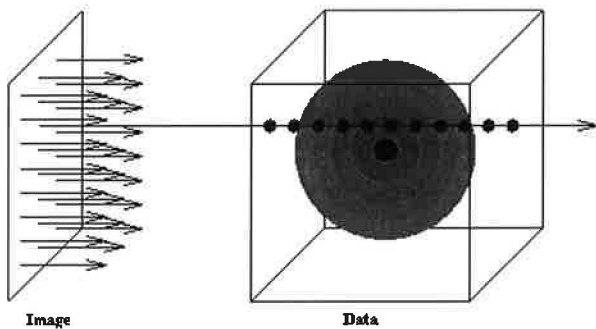
Il consiste à fournir un affichage complet de l'objet, en décrivant ses structures internes ainsi que sa surface [4]. On parle d'*affichage volumique direct* car ses données d'entrées sont les voxels eux-mêmes. Des effets de transparence peuvent être utilisés afin de laisser apparaître l'intérieur de l'objet, comme dans l'image suivante :



Vision en semi-transparence par Affichage Volumique Direct.

Les approches classiques pour ce type d'affichage sont les suivantes :

- lancer de rayons [5], [7], [8] (*ray casting*) : une valeur de couleur et une valeur d'opacité sont calculées pour chaque élément de volume. Un rayon est lancé du plan image vers l'ensemble des données, sur lequel seront échantillonnées les valeurs de couleur et d'opacité. Cette méthode est différente du lancer de rayons classique qui prend en compte les réflexions : ici un seul trajet est calculé, le rayon s'arrêtant lorsque la valeur maximale d'opacité est atteinte ou lorsqu'il sort de l'espace des données. Aucune ombre ou réflexion n'est donc produite. Les données de couleur et d'opacité sont à chaque fois pondérées selon le gradient local ;



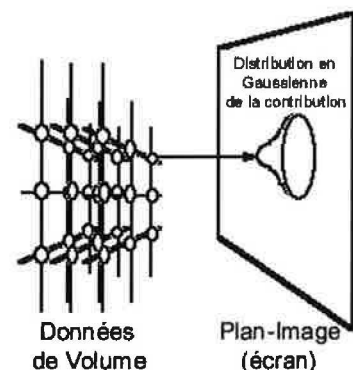
A gauche : le rayon traverse l'objet et récupère des valeurs d'opacité et de couleur. Le cheminement du rayon emploie une technique essentiellement basée sur l'algorithme de tracer de droite de *Bresenham* [18].

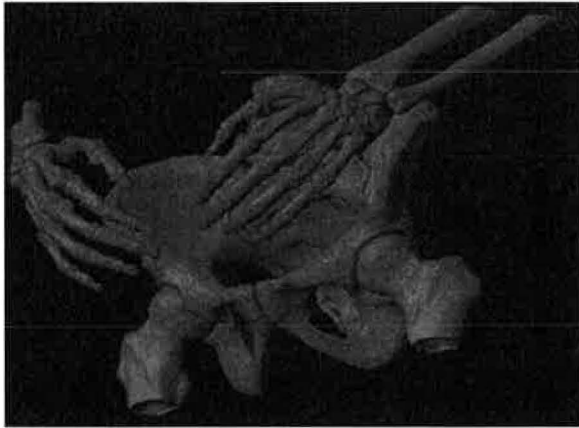
- Diffusion (*splatting* et *V-buffer*) [9], [6] : une valeur de couleur et une valeur d'opacité sont calculées pour chaque élément de volume. Ceux-ci sont ensuite projetés sur le plan de l'image : l'image finale tient alors compte de la contribution de chaque élément de volume dans chaque pixel de l'écran ;

Chaque élément de volume est traité comme une source de lumière qui émet vers le plan de l'image mais également sur les zones voisines (distribution gaussienne). Les plans sont traités séparément [10] et projetés progressivement sur le plan de l'image : on obtient ainsi une *visualisation progressive*.

Traversée des plans  
d'avant en arrière

Couleur et opacité sont  
combinées sur le plan-image





Ci-contre une image calculée par splatting (à partir du jeu de données *Visible Man*).

- Eclaté (*slicing*) : cette technique, surtout utilisée en imagerie médicale, réalise une projection classique des coupes tomodensitométriques obtenues par scanner ou IRM sur le plan de l'image ;



A droite : chaque plan est calculé indépendamment et peut être ensuite affiché sur l'écran selon des algorithmes classiques de projection.

- décalage-déformation (*shear-warping*) [19] : cette méthode est focalisée sur l'augmentation de la vitesse d'affichage. Elle emploie pour cela un algorithme dit de *factorisation-décalage-déformation* qui transforme l'objet en une représentation intermédiaire qui va accélérer les calculs.

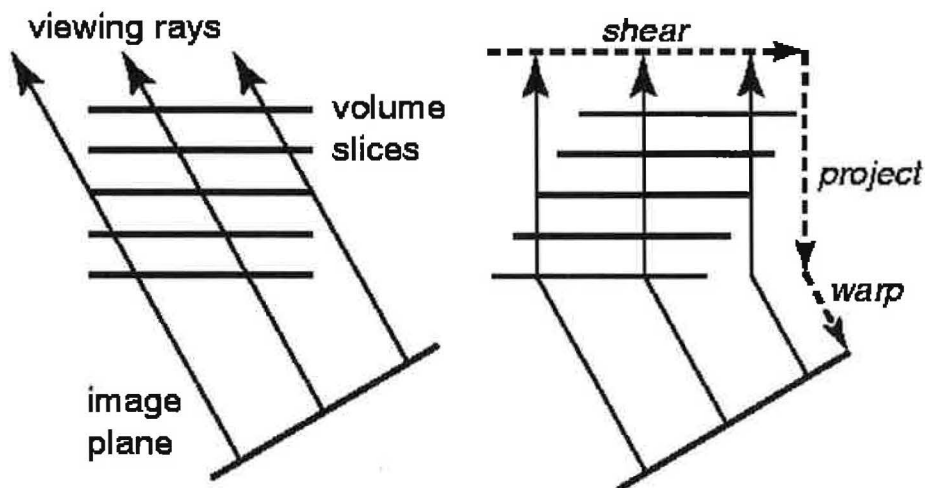


Figure 6

Le principe est de générer une configuration pour laquelle les rayons soient perpendiculaires aux plans des éléments de volume. Pour cela un décalage (*shearing*) des plans est réalisé selon le point de vue (image de droite), ainsi qu'un changement d'échelle pour recréer la perspective. Une vue est alors générée, déformée (*warp*) et enfin projetée sur le plan-image. De telles opérations sont très peu coûteuses en temps de calcul.



A gauche : quelques images générées par la technique de décalage-déformation.

Comparaison des méthodes :

- l'éclaté est le plus rapide mais met le moins en valeur l'information ;
- le lancer de rayons est de très bonne qualité. Il peut être parallélisé car chaque rayon est indépendant et des techniques d'accélération matérielle dédiées existent [17] ;
- la diffusion autorise l'affichage progressif, mais nécessite une grande puissance de calcul ;
- le décalage-déformation est la méthode la plus performante en terme de rapport qualité/temps de calcul. Elle peut bénéficier d'un système matériel dédié, même si rien n'existe à l'heure actuelle.

Avantages :

- pas d'ambiguïté topologique puisque les données sont affichées directement sans phase d'interpolation ;
- haute qualité d'image, notamment par lancer de rayons ;
- meilleure appréhension du volume par semi-transparence pour les structures internes ;

Inconvénients :

- l'aspect visuellement flou rend l'interaction cruciale afin de mieux appréhender l'information, ce qui se heurte à la puissance de calcul exigée ;
- à chaque affichage, le jeu de données complet doit être parcouru ;
- les accélérateurs matériels dédiés pour ce type de technique restent peu développés.



### 3. REPRÉSENTATION SURFACIQUE DES OBJETS.

Le mode de représentation surfacique est bien sûr employé lorsqu'une description interne d'un objet n'est pas présente, mais dans le cas contraire il autorise un affichage rapide et synthétique de celui-ci.

#### 3.1. Acquisitions surfaciques.

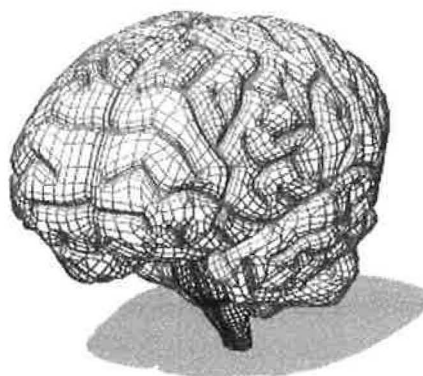
Le système de *télémétrage par laser* (*Light Amplification by the Stimulated Emission of Radiation*) consiste à projeter un rayon sur une surface solide tout en mesurant le temps entre l'émission du rayon et son observation par un capteur CCD par exemple (*Charge-Coupled Device*). (Image courtoisie du CRAI, Centre de Recherche en Architecture et Ingénierie).



Ces deux images montrent respectivement de gauche à droite l'objet archéologique réel et les résultats obtenus après acquisition par télémétrage laser : on remarque les innombrables points composant l'objet numérisé. Aucune information surfacique n'étant enregistrée, les points doivent être très resserrés pour

capturer au mieux le relief.

D'autres techniques basées sur le même principe permettent de choisir les points à mesurer : dans ce cas, chaque point indique le sommet d'un contour fermé plan ou *polygone*. Un appareil semblable à un stylo est positionné sur une maquette de l'objet aux endroits prédéfinis et l'information de position est chaque fois envoyée à l'ordinateur (image courtoisie de ViewPoint Digital). On notera la taille variable des polygones, qui sont plus petits dans les zones à fort relief, ceci afin de se rapprocher au mieux des formes originales.



© Copyright 1999 Viewpoint Digital or its licensors

### 3.2. Modèles internes surfaciques.

#### 3.2.1. La représentation polygonale directe.

Elle ne décrit explicitement que la surface de l'objet. Ses principaux défauts en tant que modèle de représentation sont les suivants :

- expression lourde de la surface ;
- pas de continuité garantie entre les facettes ;
- pas d'indépendance vis-à-vis du changement d'échelle par exemple (si l'on agrandit l'objet, les facettes deviennent d'autant plus grande, ce qui réduit la précision relative).



Ce véhicule à gauche est un objet tridimensionnel représenté de manière complètement explicite : chaque facette est décrite sous la forme d'un ensemble de coordonnées  $(x, y, z)$  désignant ses sommets.

### 3.2.2. Les fonctions explicites.

C'est le cas le plus simple de représentation analytique surfacique :

Pour chaque valeur de  $x$  et  $y$ , il existe une valeur  $z$  ( $z=f(x,y)$ ) unique correspondante : c'est pour cela que l'on parle parfois de «  $2D\frac{1}{2}$  » au lieu de 3D.

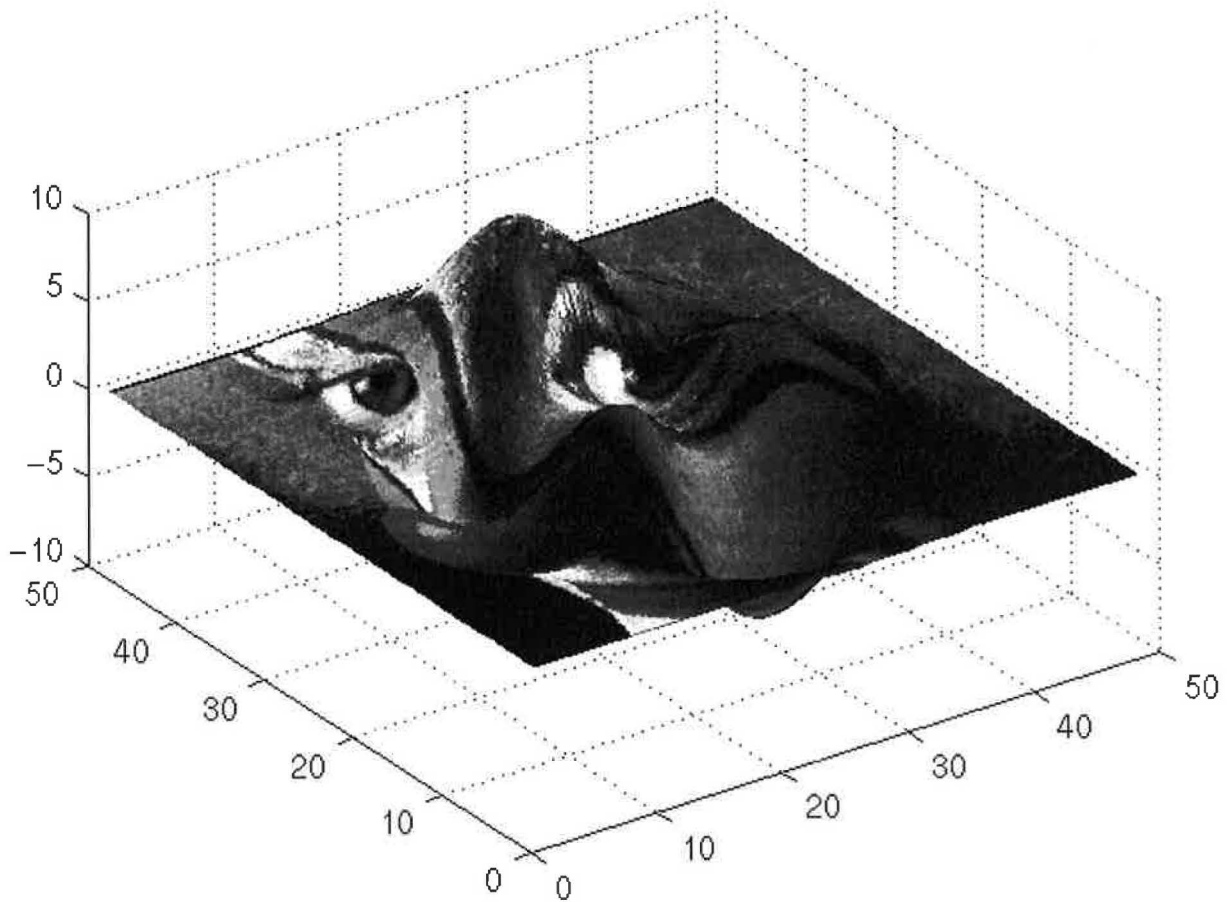


Figure 7

Surface tracée avec le logiciel *Matlab* de *MathWorks*, sur laquelle a été projetée une texture. Les données proviennent d'une fonction évaluée en chaque point  $(x,y)$  de la grille, déterminant ainsi l'élévation  $z$ .

Il est impossible de représenter des courbes fermées, ou exhibant plusieurs valeurs pour un même  $x$ .

### 3.2.3. La représentation paramétrique.

Elle consiste à faire dépendre les coordonnées d'une variable  $t$  appelée *paramètre*.  
 Considérons l'exemple 2D suivant :

$$\begin{aligned}x &= x_0 + r \cdot \cos(t) \\ y &= y_0 + r \cdot \sin(t)\end{aligned}$$

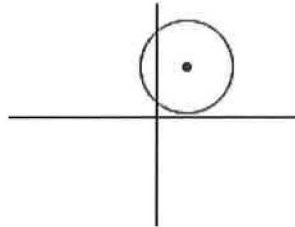


Figure 8

L'ensemble de ces deux équations décrit un cercle dans le plan de centre  $(x_0, y_0)$ , de rayon  $r$ , sachant que  $t$  varie entre 0 et  $2\pi$ .

Il en est de même en trois dimensions, où l'on introduit un second paramètre pour décrire cette fois-ci une surface. On note alors usuellement  $u$  et  $v$  les deux paramètres.

$$\begin{aligned}x &= x_0 + r \cdot \sin(u) \cdot \sin(v) \\ y &= y_0 + r \cdot \sin(u) \cdot \cos(v) \\ z &= z_0 + r \cdot \cos(u)\end{aligned}$$

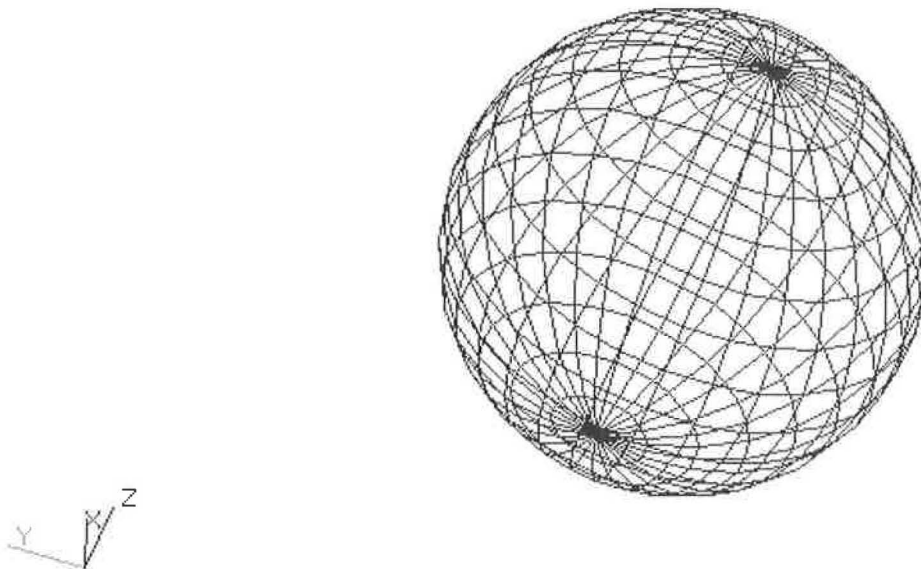
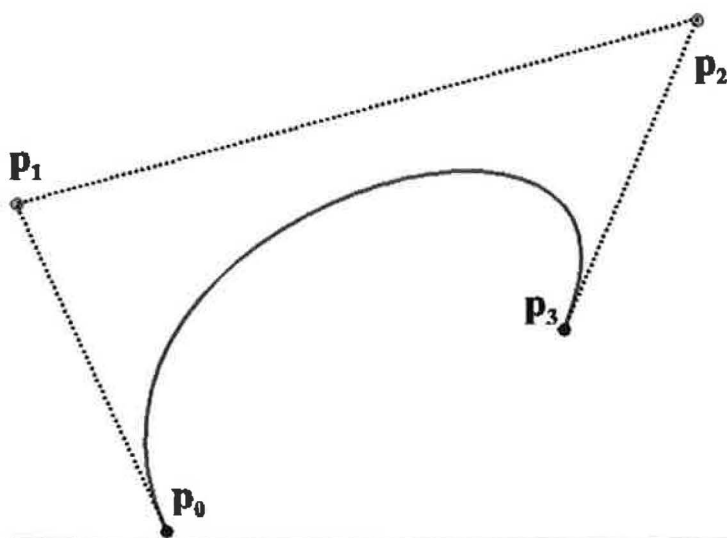


Figure 9

Equation paramétrique d'une sphère ( $u$  et  $v$  varient entre 0 et  $2\pi$ ).

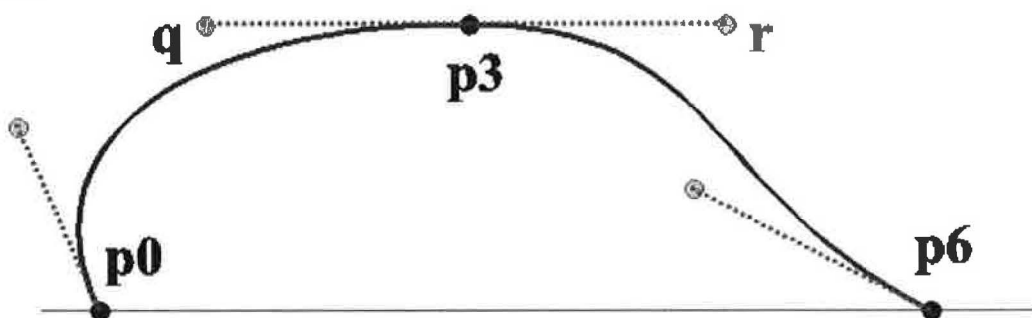
Ce mode de représentation est très largement répandu puisqu'il est employé dans le domaine de la CFAO (*Conception et Fabrication Assistée par Ordinateur*). Ce besoin est en effet très spécifique aux professions d'architecte et d'ingénieur, et date du commencement de l'informatisation. Avant cela, les objets nécessitant des parties courbes comme les coques de bateaux étaient facilement représentables sur une planche à dessin mais pas du tout en taille réelle ou quasi-réelle. Pour réaliser une telle tâche on employait de longues lattes en bois, plastique ou métal appelées *splines* dont la flexion était réglée par un ensemble de poids. Les courbes ainsi obtenues étaient continues, la courbure dépendant de la position et de l'intensité des poids. Lorsque l'ordinateur fit son apparition, les propriétés physiques de ces lattes furent étudiées et modélisées [24], donnant naissance aux *courbes splines*. Un des pionniers fut Pierre Bézier, ingénieur chez le constructeur automobile français Renault, qui développa en 1970 le modèle des *courbes de Bézier* [25].



Formellement, ces « lattes » sont modélisées par des fonctions polynomiales par morceau, raccordées selon des contraintes de continuité particulières ( $C^1$ ,  $C^2$ ...). La courbe présentée à gauche (dite *de Bézier*) est modélisée par le polynôme suivant :

$$x(t) = p_0 (1-t)^3 + 3 p_1 t (1-t)^2 + 3 p_2 t^2 (1-t) + p_3 t^3$$

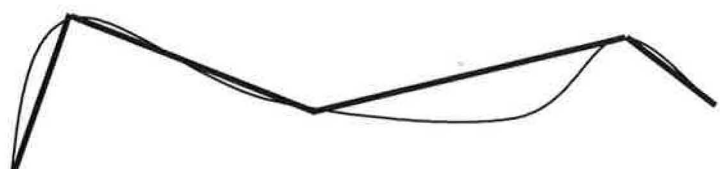
où  $t$  est le paramètre décrivant la courbe. Cette courbe peut être raccordée à d'autres comme nous le montre la figure suivante :



Le raccord en  $p_3$  (aussi appelé *nœud*) sur cette spline est réalisable selon plusieurs méthodes (ici technique de Catmull-Rom avec tension).

Un avantage important des courbes paramétriques repose sur le pas de discrétisation du paramètre  $t$  : son amplitude définit une représentation naturelle par niveaux de détails :

La ligne épaisse (image de droite) désigne un pas de  $t$  assez grand, qui



est contrôlable selon la précision désirée. La spline sous-jacente utilise un pas beaucoup plus fin.

On distingue aujourd'hui les familles suivantes :

- courbes de Bézier :

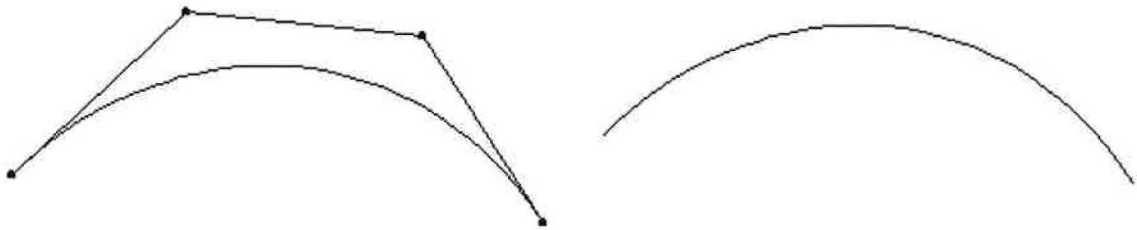
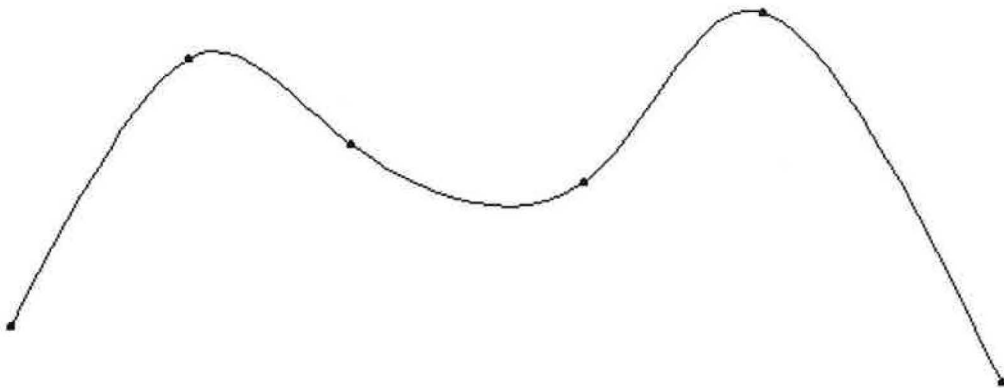


Figure 10

On remarque à gauche l'ensemble des points de contrôle permettant d'agir sur la courbe. Une propriété intéressante d'une telle courbe est qu'elle passe par les points de contrôle situés aux extrémités. L'image de droite montre la courbe résultante.

- Splines d'interpolation [26], [27] [28] :

Entre chaque couple de points une spline cubique (de degré trois) est positionnée, les contraintes de continuité étant localisées en chaque point.



Cette courbe passe par ses points de contrôle, d'où son nom de *spline d'interpolation*. Dans le cas contraire, on dit *spline d'approximation* (cas suivants).

- B-splines (splines d'approximation) [29], [27] :

Elles présentent la propriété d'être déformables de manière locale :

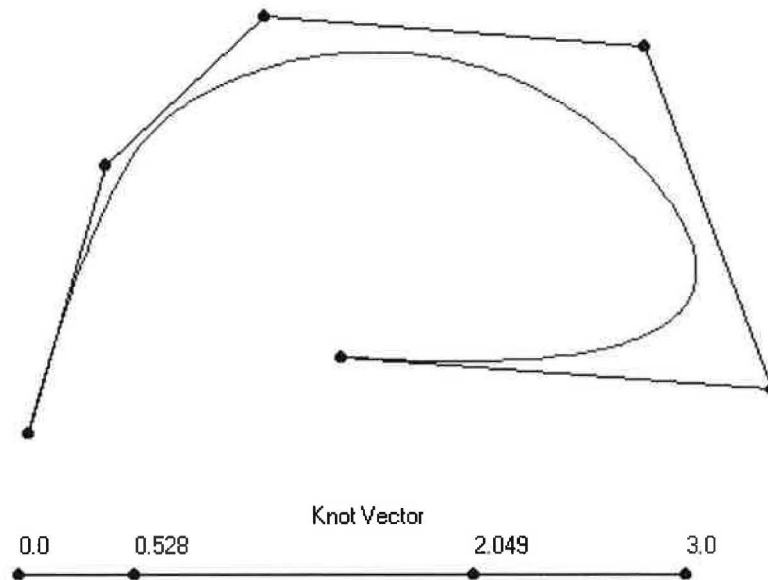


Figure 11

Cette B-spline (en bleu) est définie par un ensemble de points de contrôle dont on peut régler l'influence le long de la courbe (*knot vector*), c'est-à-dire la portée de la déformation. A noter que normalement la courbe ne passe par aucun point de contrôle : les extrémités de la spline sont soumises à une forte pondération des points de contrôle et sont ainsi « attirées » vers ces points.

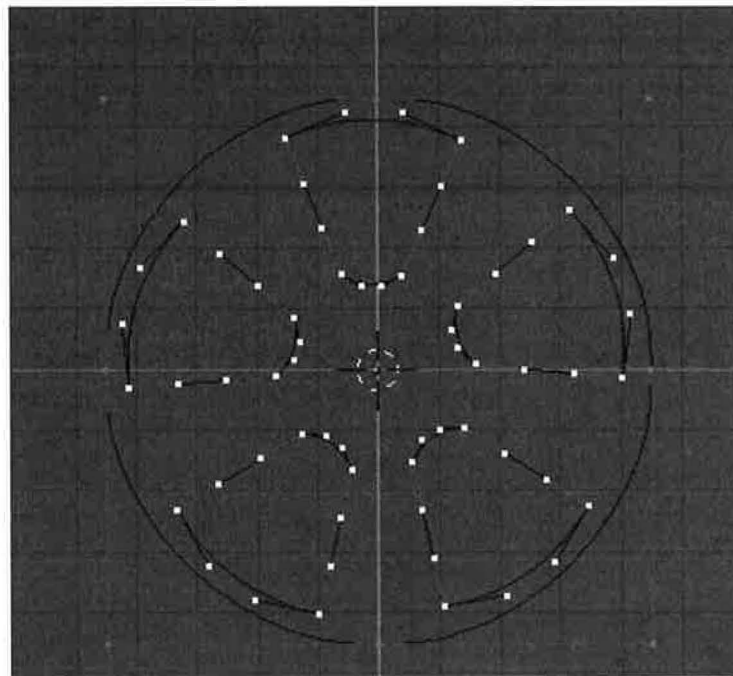


Figure 12

Voici un exemple de création d'une jante de voiture et d'un profil de roue en employant des NURBs (*Non-Uniform Rational B-spline*). On distingue les points de contrôle et les tangentes contraignant les splines. (Image réalisée avec le logiciel *Blender*.)

- Bêta-splines (approximation) [30] :

Elle sont dotées de deux paramètres de contrôle supplémentaires par rapport aux B-splines, qui sont respectivement le *biais* et la *tension* et qui agissent sur la méthode d'approximation. A partir de ces courbes il devient possible de générer des surfaces (ou *patches* en anglais) par *produit tensoriel*. La méthode d'interpolation de *Coons* [31] propose une autre méthode de génération de surfaces à partir de quatre bords.

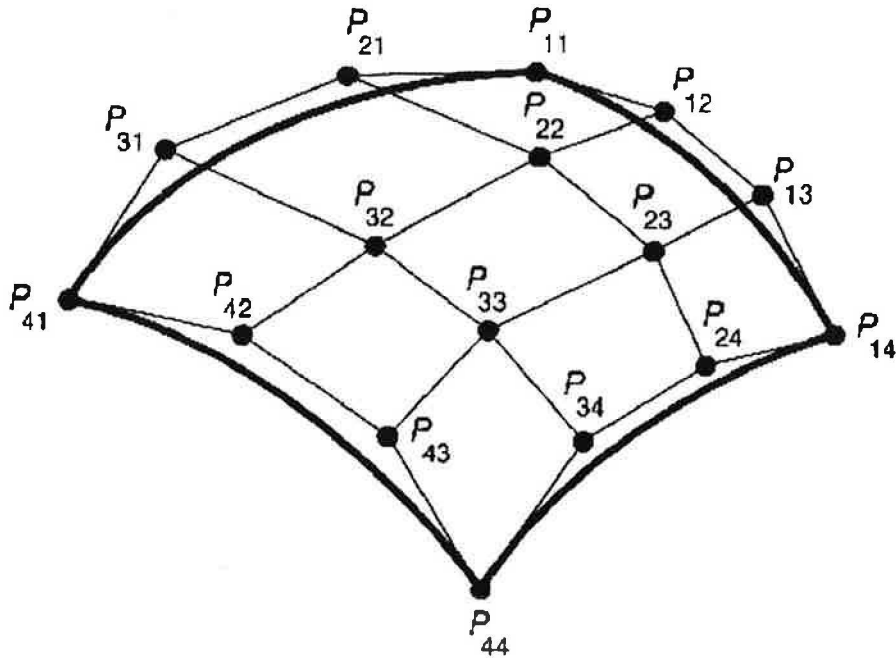


Figure 13

Cette surface est définie en utilisant des points de contrôles fonctionnant de la même manière que les courbes de Bézier en 2D : au-dessus de la surface proprement dite on peut voir le réseau de contrôle. Ce type de surface est nommé *carreau (patch)*.

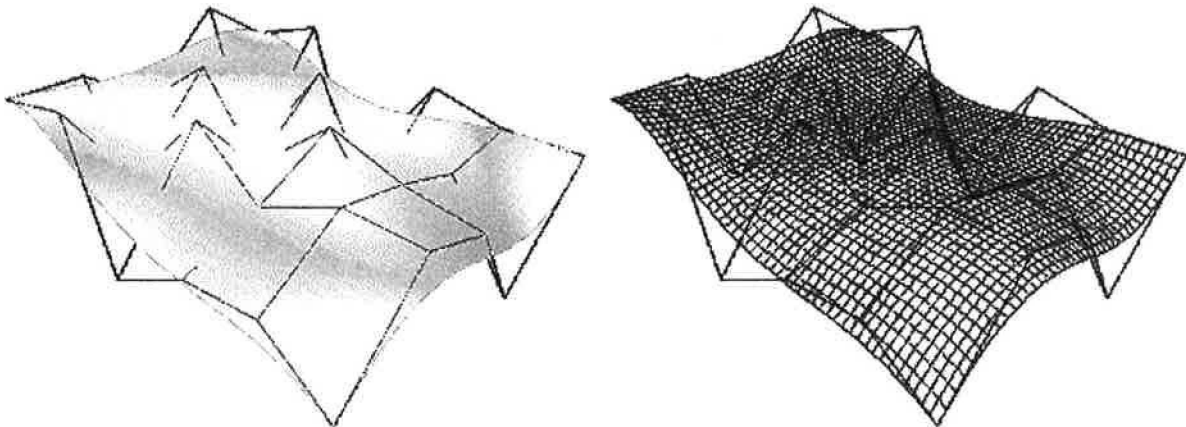


Figure 14



## 2. Tour d'Horizon de la Représentation Tridimensionnelle.

---

Cette surface est générée à partir de carreaux de Bézier. Des contraintes de continuité sont mises en place entre chaque carreau de manière à ce que la surface totale soit elle-même continue.

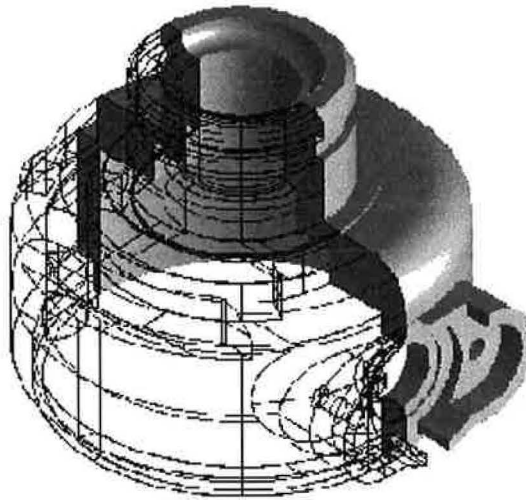


Figure 15

Voici une pièce mécanique modélisée par courbes NURBs en trois dimensions. (Image courtoisie de *CIMATRON USA*).

Avantages :

- des formes complexes peuvent être décrites simplement et intuitivement.

Inconvénients :

- les contraintes de continuité entre carreaux contigus peuvent être difficiles à maintenir (cas des B-splines) ;
- elles sont complexes à mettre en œuvre et coûteuses en temps de calcul (*patches* de Bézier-Gregory par exemple) ;
- les calculs d'intersection ne sont pas efficaces : un ensemble de surfaces ne décrit pas forcément la frontière d'un objet 3D et vérifier ce fait s'avère délicat ;
- il est difficile de déterminer si un point donné est à l'intérieur ou à l'extérieur de l'objet dont on a décrit la frontière par de telles surfaces.

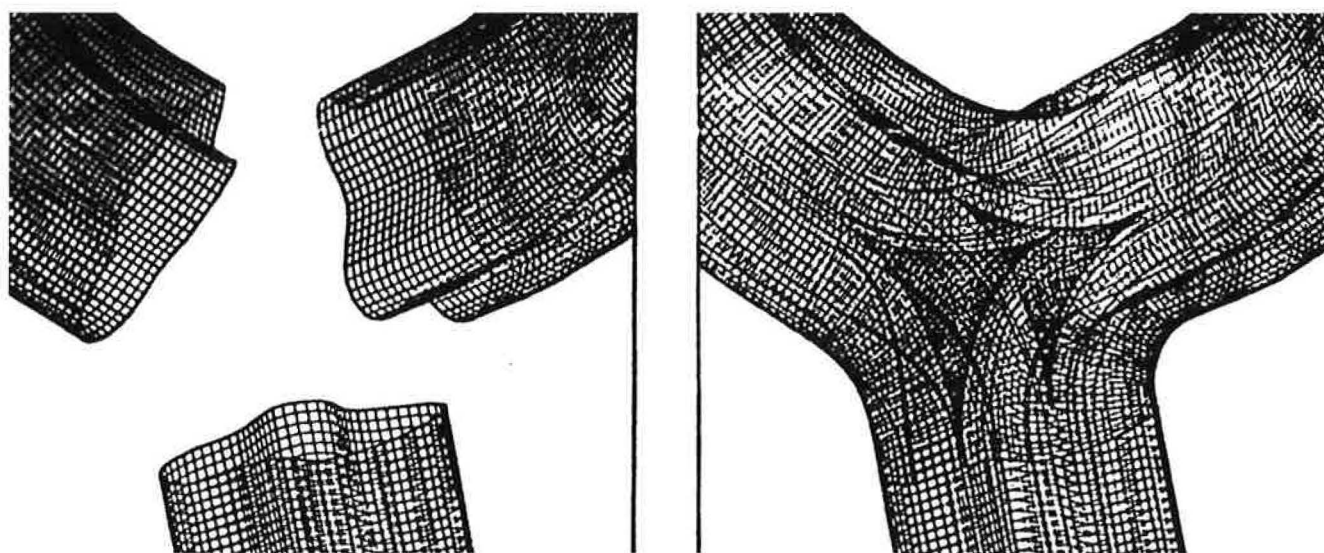


Figure 16

La jonction des ces tubes paramétriques est réalisée (à droite) par une technique nommée « *patch de Gregory* » ou « *Bézier-Gregory* ». Cette technique requiert des degrés de polynômes élevés et des contraintes de continuité difficiles à mettre en place (définition de plusieurs « sous parties » pour boucher les trous). (Image Courtoisie de René Ebel)

### 3.3. L'allègement des données par simplification surfacique

La représentation surfacique peut être contrôlée afin d'éviter une charge de calcul trop lourde. Cette technique, appelée *simplification surfacique* est rendue possible par l'aspect semi-continu des polygones ; ceux-ci représentant des portions de plan, leur nombre peut être réduit tout en conservant un aspect « lisse » contrairement aux voxels qui ne peuvent être décimés (ce qui donne alors des résultats très grossiers en termes de géométrie).

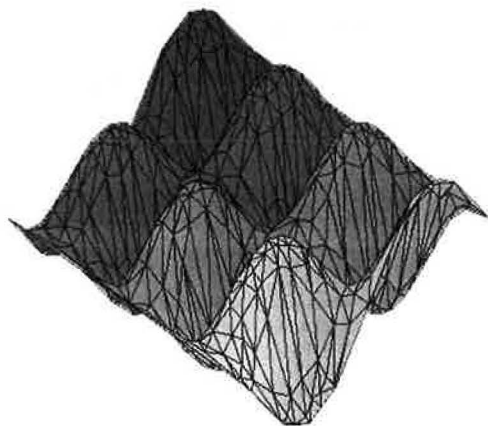
De manière générale, le processus de simplification surfacique répond aux problèmes suivants :

- accélérer l'affichage et par voie de conséquence améliorer l'interaction ;
- accélérer les transferts par réseau ;
- diminuer l'occupation en mémoire de masse et éventuellement en mémoire vive.

## 2. Tour d'Horizon de la Représentation Tridimensionnelle.

La simplification peut être opérée de différentes manières, aussi distingue-t-on deux classes de techniques :

- la génération *adaptive* de surfaces polygonales [21], c'est-à-dire tenant compte du relief de l'objet. Il s'agit en d'autres termes d'utiliser plus de polygones dans les zones accidentées.

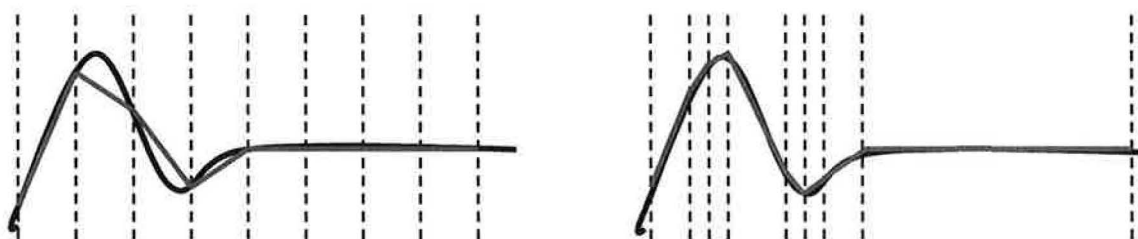


A gauche : la taille des polygones est dépendante du relief. On remarque de grandes facettes dans les zones planes, alors que les sommets, plus arrondis, bénéficient de mailles plus serrées.

Dans le cas d'un objet analytique, donné par une formule mathématique, les valeurs de gradient peuvent être connues en chaque point (sous contrainte de dérivabilité), ce qui autorise la régulation de la taille des polygones en fonction de la pente locale [11], [20].

Si ce n'est pas le cas, on emploie un *critère adaptatif* qui peut être :

- l'angle diédral entre deux polygones adjacents ; une trop grande valeur peut être un indice de mauvaise approximation de la surface ;
- une forte différence de l'orientation des normales aux sommets d'un polygones (donc forte variabilité surfacique dans le voisinage) ;
- une grande distance entre le centre du polygone et la surface.



A gauche : la grille est régulière et ne tient pas compte de la courbure de l'objet. Dans ce cas le relief est mal représenté. Si l'on veut améliorer la qualité, le nombre de polygones requis augmentera fortement du fait de l'affinage de la grille.

A droite : la grille est régulée selon la courbure locale [13]. De ce fait, les polygones dans les zones à fort relief sont plus nombreux mais les zones planes sont par contre très peu consommatrices de facettes. Un avantage de cette méthode est également la prise en compte naturelle des petits détails comme les bosses et les creux.

Ce type de méthode demande un temps de calcul plus élevé puisque la surface doit être explorée beaucoup plus minutieusement, ce qui entraîne des subdivisions ou des évaluations répétées de formules souvent longues à calculer.

- la *décimation* polygonale [1], [14], [12], [23] : elle consiste à ôter des polygones à un maillage existant tout en préservant son aspect visuel le mieux possible. Cette opération peut être réalisée a priori afin de générer une collection de *niveaux de détails*, ou à la volée durant l'affichage, ceci en fonction du contexte. Elle a l'avantage d'être relativement rapide, même si elle nécessite au départ un maillage très fin et qu'elle ne dépend plus de l'objet lui-même mais du maillage de départ.

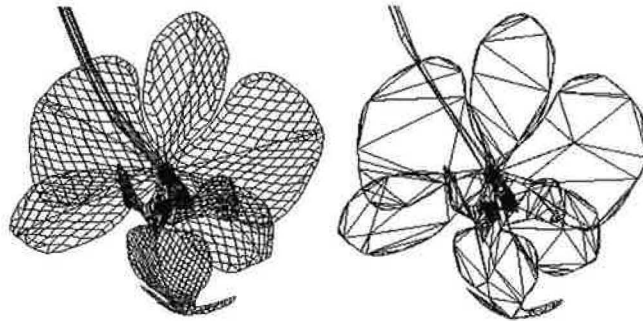


Figure 17

Simplification polygonale d'une Orchidée [1] (Maillage courtoisie de René Lecoustre / CIRAD) : la maquette simplifiée (à droite) est stockée de manière permanente sur une mémoire de masse et peut-être restaurée immédiatement durant l'affichage. Le maillage de départ est montré à gauche.

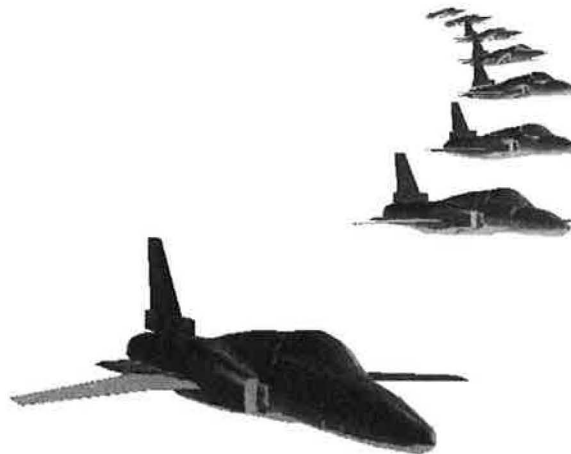


Figure 18

Dans cet exemple [1], une collection d'objets de plus en plus simples est générée, le processus d'affichage choisissant alors la représentation la plus économique tout en étant visuellement satisfaisante.

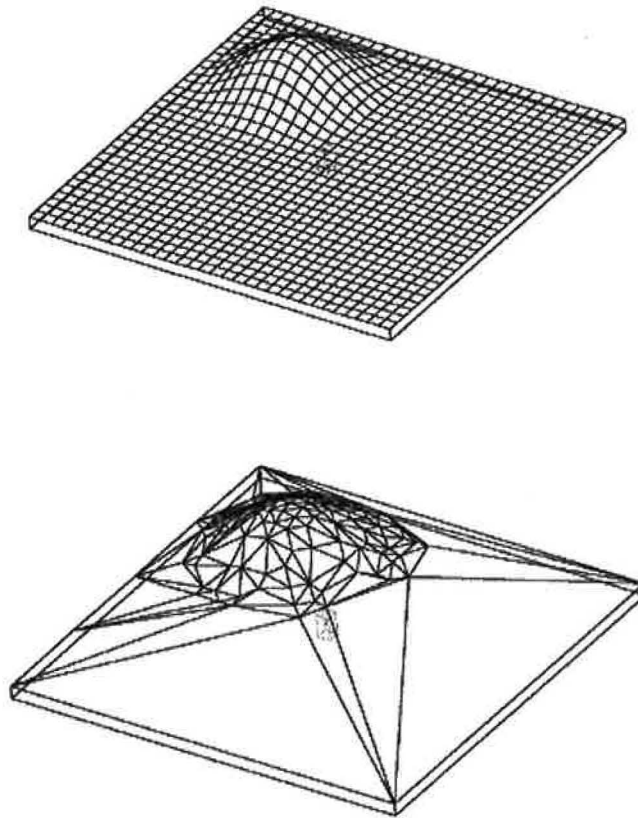


Figure 19

Ce terrain tridimensionnel est d'abord maillé régulièrement. Un algorithme examine ensuite l'angle entre les facettes et les fusionne si la valeur est trop faible.

Les algorithmes de simplification surfacique posent plusieurs problèmes, à savoir :

- la gestion des changements de topologie, comme la fusion de deux entités ou la suppression de trous topologiques dans un objet ;
- l'émergence ou la disparition soudaines de détails visuels remarquables (angles vifs, perte de détail de texture...etc.) lorsque l'objet se rapproche ou s'éloigne. Le processus de décimation doit conserver au mieux l'impression visuelle de l'objet, en offrant par exemple une simplification continue et sans heurts.

#### **4. CONCLUSION.**

Les techniques de représentation et de projection d'objets tridimensionnels comprennent au sein de chaque famille une multitude de variantes, révélatrice de l'existence de problèmes parfois difficiles. Les contraintes primordiales qui doivent guider les choix sont les suivants :

- le rapport qualité de visualisation/interaction, qui dépend de la nature de l'application ;
- la contrainte de précision (ou de réalisme) exigée par l'application ;
- le volume des données en mémoire (mémoire vive ou mémoire de masse) : ces ressources dépendent du matériel employé et plus précisément des contraintes économiques sous-jacentes ;
- la nature des données en entrée, qui dépend des modalités d'acquisition.

Il reste que la représentation par polygones est la plus largement diffusée car elle propose une expression compacte de l'objet d'une part, et qu'elle bénéficie largement de l'accélération matérielle d'autre part. Ce second point, loin d'être négligeable, autorise par exemple la visualisation en transparence des données (comme pour le rendu volumique direct) ou encore avec une finesse de représentation importante, puisque la chaîne graphique supporte sans cesse des bandes passantes de plus en plus grandes.

## 5. BIBLIOGRAPHIE.

- [1] F. Banégas, « Zoom Sémantique par Maillage Progressif », rapport de stage de DEA, CIRAD/ENSMSE, 1997
- [2] P. J. Frey, H. Borouchaki, « Texel : Triangulation de Surfaces Implicites. Partie I : Aspects Théoriques », RR N°3066, INRIA, 13 dec 1996
- [3] P. J. Frey, H. Borouchaki, « Texel : Triangulation de Surfaces Implicites. Partie II : Exemple d'Applications », RR N°3066, INRIA, 13 dec 1996
- [4] J. Wilhelms, A. V. Gelder, « A Coherent Projection Approach for Direct Volume Rendering », pp. 275-284, Computer Graphics, Proceedings of SIGGRAPH'91, Volume 25, Number 4, July 1991
- [5] H.K. Tuy, L. T. Tuy, « Direct 2D Display of 3D Objects », pp. 29-33, IEEE Computer Graphics and Application, Volume 4, Number 10, October 1984
- [6] C. Upson, M. Keeler, « The V-Buffer : Visible Volume Rendering », pp. 59-64, Computer Graphics, Proceedings of SIGGRAPH'90, Volume 22, Number 4, July 1990
- [7] M. Levoy, « Volume Rendering, A Hybrid Ray Tracer for Rendering Polygon and Volume Data », pp. 33-40, IEEE Computer Graphics and Applications, Volume 10, Number 2, March 1990
- [8] P. Sabella, « A Rendering Algorithm for Visualizing 3D Scalar Fields », pp. 51-58, Computer Graphics, Volume 22, Proceedings of SIGGRAPH'88, Number 4, July 1988
- [9] P. Hanrahan, D. Laur, « Hierarchical Splatting : A Progressive Refinement Algorithm for Volume Rendering », pp. 285-288, Computer Graphics, Volume 25, Number 4, Proceedings of SIGGRAPH'91, August 1991
- [10] L. Westover, « Footprint Evaluation for Volume Rendering », pp. 367-376, Computer Graphics, Volume 24, Number 4, Proceedings of SIGGRAPH'90, August 1990
- [11] J. P. Thirion, A. Gourdon, « Computing the Differential Characteristics of Isodensity Surfaces », pp. 190-202, CVGIP, 1995.
- [12] W. J. Schroeder, J.A. Zarge and W. Lorensen, « Decimation of Triangle Mesh », pp. 65-70, ACM Computer Graphics, Proceedings of SIGGRAPH'92, Volume 26, Number 2, 1992
- [13] M. S. Sheppard, M. K. Georges, « Automatic Three Dimensional Mesh Generation by the Finite Octree Technique », pp.709-749, Int. J. Numer. Mthods Eng., Volume 32, 1991
- [14] H. Hoppe, « Progressive Meshes », pp. 99-108, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996
- [15] P. Lindstrom, D. Koller, W. Ribarsky, L. F. Hodges, N. Faust and G. A. Turner, « Real-Time, Continuous Level of Detail Rendering of Height Fields », pp. 109-118, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996
- [16] J. Cohen, A. Varshney, D. Manocha, G. Turk, H. Weber, P. Agarwal, F. Brooks and W. Wright, « Simplification Envelopes », pp. 119-128, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996



- [17] H. Pfister, J. Hardenbergh, J. Knittel, H. Lauer and L. Seiler, « The VolumePro Real-Time Ray Casting System », pp. 251-260, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH99, 1999
- [18] T. Pavlidis, « Algorithms for Graphics and Image Processing », Springer 1982
- [19] R. Drebin, L. Carpenter and P. Hanrahan, « Volume Rendering », pp. 60-69, Computer Graphics, SIGGRAPH88, Volume 22, Number 4, August 1988
- [20] M. Do Carmo., « Differential geometry of curves and surfaces », Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1976.
- [21] N. Dyn, D. Levin, and S. Rippa., « Algorithms for the construction of data dependent triangulations », Algorithms for Approximation II, J.C. Mason and M.g.Cox, eds. Chapman and Hall, NY, 10(5):185-192, 1990.
- [22] J. Bloomenthal, K. Ferguson, « Polygonization of Non-Manifold Implicit Surfaces », pp. 309-316, ACM Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH'95, 1995
- [23] J. Popovic, H. Hoppe, « Progressive Simplicial Complexes », pp. 217-224, ACM Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH'97, 1997
- [24] M. A. Sabin, « *Parametric Splines in Tension* », Technical Note VTO/MS/160, British Aircraft Corporation, Weybridge, Surrey, England, July 1970
- [25] P. E. Bézier, « *Essai de Définition Numérique des Courbes et des Surfaces Expérimentales* », PhD dissertation, Université Pierre et Marie Curie, Paris, February 1977
- [26] J. D. Foley, A. van Dam, « *Fundamentals of Interactive Computer Graphics* », Addison-Wesley, Reading, Massachusetts, 1982
- [27] W. M. Newman, R. F. Sproull, « *Principles of Interactive Computer Graphics* », second Edition, McGraw-Hill, New-York
- [28] L. L. Schumaker, « *Spline Functions : Basic Theory* », John Wiley & Sons, New-York, 1981
- [29] R. H. Bartels, J. C. Beaty, B. A. Barsky, « *B-splines* », Mathématiques et CAO, vol. 6, Morgan Kaufmann Publishers, 1987, Hermès Publishing, 1988
- [30] R. H. Bartels, J. C. Beaty, B. A. Barsky, « *Bêta-splines* », Mathématiques et CAO, vol. 7, Morgan Kaufmann Publishers, 1987, Hermès Publishing, 1988
- [31] S. A. Coons, « *Surfaces for Computer Aided Design* », Projet MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, 1964
- [32] J. F. Blinn, « *A Generalization of Algebraic Surface Drawing* », ACM Transactions on Graphics, 1(3), July 1982
- [33] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa and K. Omura, « *Object Modeling by Distribution Function and a Method of Image Generation* », Journal of Electronic Communications, 1985
- [34] G. Wyvill, C. McPheeters and B. Wyvill, « Data Structures for Soft Objects », The Visual Computer, 2(4), August 1986
- [35] S. Muraki, « Volumetric Shape Description of Range Data using « Blobby Model » », Computer Graphics, 25(4), pp. 227-235, July 1991
- [36] T.A. Galyean and J.F. Hughes. « *Sculpting : An interactive volumetric modeling technique.* » Computer Graphics, 25(4) :267–274, July 1991. Proceedings of SIGGRAPH'91 (Las Vegas, Nevada, July 1991).



- [37] S.W. Wang and A.E. Kaufman. « *Volume sculpting* ». Computer Graphics, pages 151–156, 1995. Proceedings, Symposium on Interactive 3D graphics.
- [38] E. Ferley, M-P. Cani, J-D. Gascuel, « *Practical Volumetric Sculpting* », Proceedings of the Fourth International Workshop on Implicit Surfaces'99, pp. 51-58, ACM Press, Talence, France, 13-15 September 1999

## **3. Tour d'Horizon de la Reconstruction Surfactive.**

## 1. INTRODUCTION.

Afficher un objet tridimensionnel défini seulement par un ensemble de points non connectés requiert la reconstruction de sa géométrie (selon un des modèles décrits dans le chapitre traitant de la représentation tridimensionnelle). Le problème se rencontre notamment en CAO (Conception Assistée par Ordinateur), où différents capteurs permettent d'échantillonner les surfaces des objets : on souhaite ensuite reconstruire ces surfaces pour les intégrer dans les logiciels de CAO usuels d'usinage, de calculs de contraintes...Etc. Dans le monde médical actuel le besoin émerge, par exemple pour conférer ensuite aux objets reconstruits des propriétés mécaniques et les manipuler interactivement avec une qualité visuelle élevée. Dans le cas de la CAO, le nuage de points est souvent non structuré, alors qu'en médecine les points sont typiquement placés sur une grille cubique régulière (ou anisotrope dans une direction particulière). En fait, le problème de la reconstruction surfaccique se rencontre aussi quand une définition analytique de l'objet est disponible (par surface implicite par exemple). Il n'est en effet pas évident de détecter la surface de tels objets ou même de déterminer si l'objet décrit bien un volume borné.

Nous allons ici nous intéresser plus précisément aux moyens mis en œuvre pour la reconstruction surfaccique. Nous étudierons également dans ce chapitre quelques techniques d'optimisation numériques nécessaires pour guider le processus de reconstruction dans certains cas.

La figure suivante met en exergue la profonde rétroaction entre le modèle qui représente l'objet et la méthode qu'il faudra employer pour reconstruire explicitement sa surface :

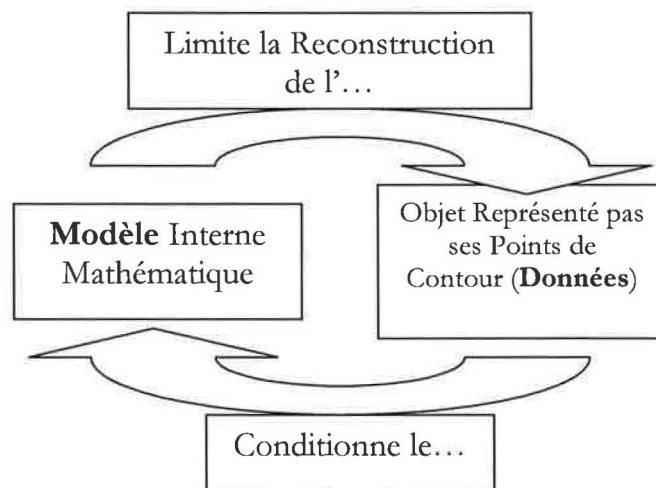


Figure 1

Une rétroaction qui rend les décisions difficiles...

Ce que l'on attend d'un bon modèle de représentation surfactive regroupe les critères suivants :

- il autorise une bonne précision dans la géométrie ;
- l'expression de la surface est la plus concise possible ;
- il autorise des modifications locales (déformation) ;
- il gère des topologies arbitraires ;
- sa visualisation est rapide ;
- les calculs d'intersection entre surfaces sont efficaces.

**2. LA RECONSTRUCTION SURFACIQUE.**

Elle consiste essentiellement à faire correspondre un modèle explicite de surface avec les données définissant les contours d'un objet 3D. Ces données revêtent divers aspects : nuage de points, équation, segments sur plusieurs coupes...etc. La qualité du résultat se mesure par l'erreur commise entre la surface reconstruite et la surface « réelle » de l'objet.

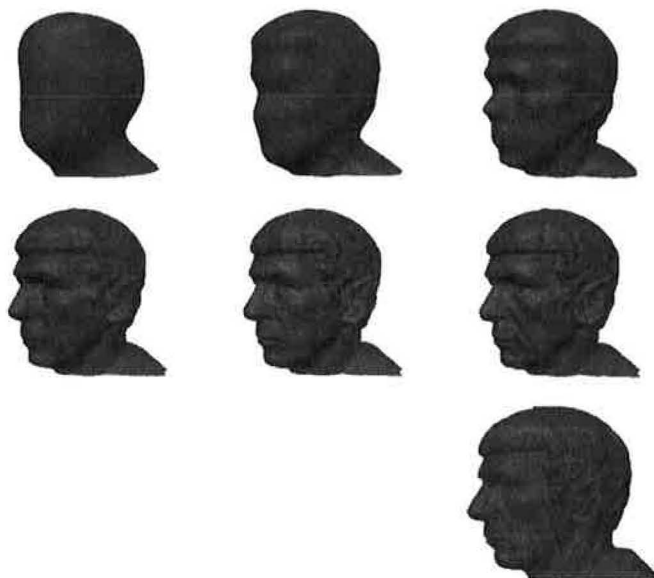


Figure 2

La Figure 2 montre les différentes étapes de la reconstruction de ce visage déjà rencontré précédemment. La surface est représentée par des carreaux de surfaces paramétriques, ce qui explique la finesse des détails. L'image en bas à droite (en gris) montre l'ensemble des points tridimensionnels à approcher. Ce procédé s'inspire de travaux présentés en [24].

On peut qualifier le procédé de reconstruction surfactive d'*ingénierie inverse* puisque la donnée de départ est l'objet « réel » et la donnée d'arrivée est la structure de données définissant une surface (dans notre cas). Comme nous allons le découvrir, les techniques d'obtention de surface, quoique déclinées en de nombreuses variantes, appartiennent essentiellement à deux catégories :

- le placement de primitives ou de squelettes de manière *procédurale* au sein du nuage des points de contour. Nous entendons par primitives des objets géométriques tels que la sphère, le cube mais aussi le polygone, le segment...Etc. ;
- la recherche de la meilleure surface approchante par *optimisation numérique*.

La première catégorie consiste le plus souvent à créer une représentation explicite de la surface à partir des seuls points de contour. La seconde nécessite l'expression analytique d'une première approximation de l'objet, sur laquelle elle va agir pour converger vers les points de contour. La Figure 3 illustre cet état de fait :

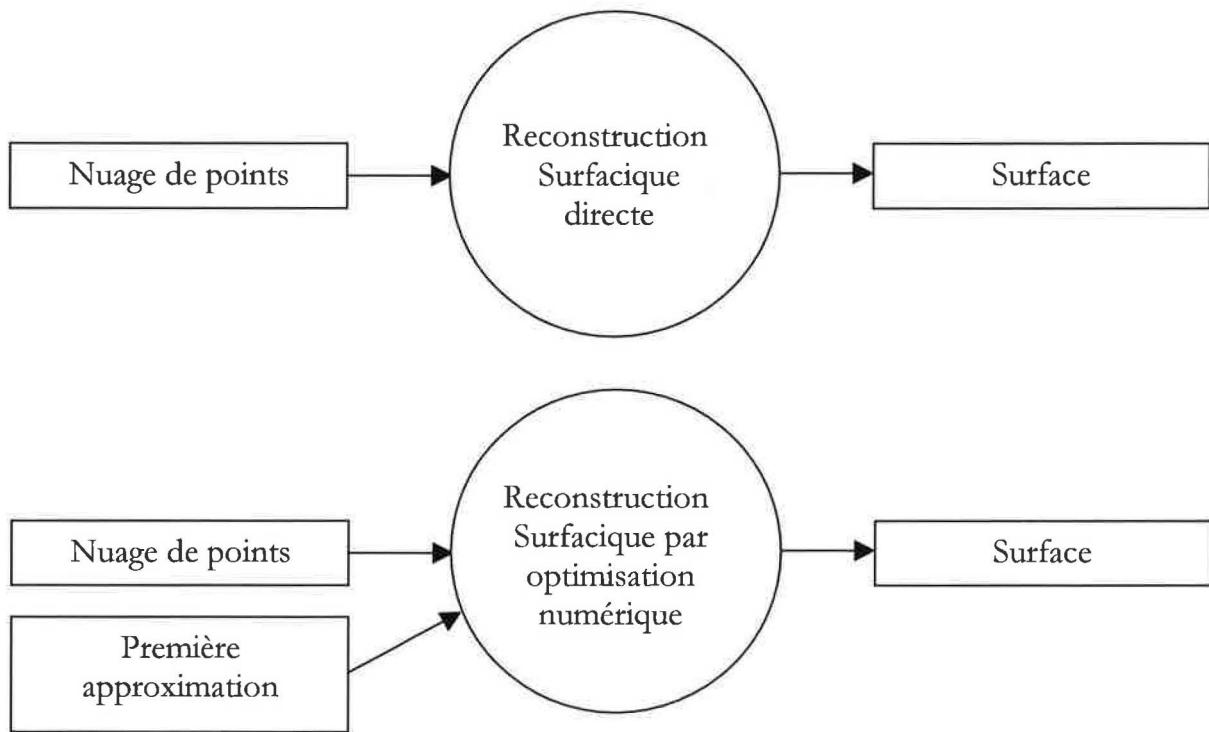


Figure 3

Les principales difficultés inhérentes aux procédés de reconstruction surfaccique sont dues à la qualité des données numériques. Celles-ci peuvent en effet être altérées de différentes façons :

- elles peuvent être bruitées, ce qui est essentiellement dû à la précision de l'appareil de numérisation ;
- elles peuvent être incomplètes (occlusions lors de la numérisation) ;
- leur densité peut ne pas être uniforme – c'est le cas par exemple des surfaces verticales échantillonnées par palpeurs ;
- certaines informations géométriques peuvent être erronées à cause d'un défaut dans l'algorithme de numérisation.

### 2.1. Reconstruction surfaccique directe.

La reconstruction surfaccique procédurale est un algorithme destiné à reconstruire explicitement la surface d'un objet tridimensionnel. Si l'objet est déjà exprimé sous forme de polygones, cette méthode se réduit à l'identité. Dans le cas contraire elle peut agir de deux manières différentes :

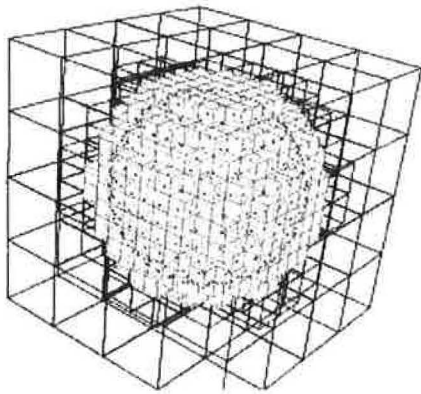
- construire directement un ensemble de polygones approchant la surface. C'est ce que nous nommerons *construction directe* (section 2.1.1) ;

- fournir une expression analytique ou procédurale permettant par la suite d'obtenir l'ensemble de polygones de manière directe comme indiqué dans le point précédent. C'est la *construction indirecte* (section 2.1.2).

Le lecteur notera que cette phase peut être opérée en amont, constituant alors une donnée d'entrée pour une reconstruction surfactive par optimisation numérique [5].

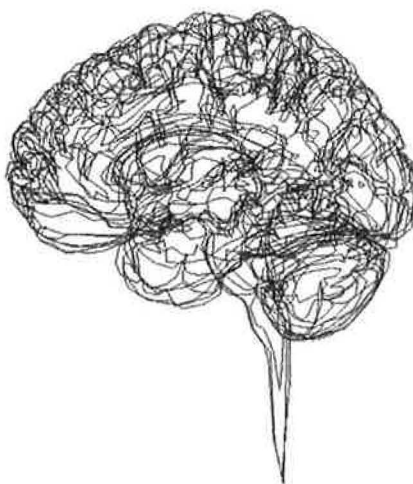
### 2.1.1. Représentation explicite du contour : construction directe.

Si l'objet est exprimé de manière analytique, il devient possible de mettre en correspondance sa surface avec des primitives planes (polygones). Ces contours, définissables de plusieurs manières (fonctions implicites, isovaleur, fonctions paramétriques...etc.), sont plongés dans une grille régulière tridimensionnelle [28], [29] : les intersections entre ces contours et la grille serviront de points de base pour construire la surface polygonale.



Exemple d'immersion d'un objet (la sphère) dans une grille discrète tridimensionnelle. Notons qu'une grille n'est pas forcément régulière comme c'est le cas ici et qu'elle peut être adaptée à la géométrie ou à la topologie de l'objet.

S'il est exprimé de manière discrète (points, contours...etc.) les techniques suivantes s'appliquent alors :

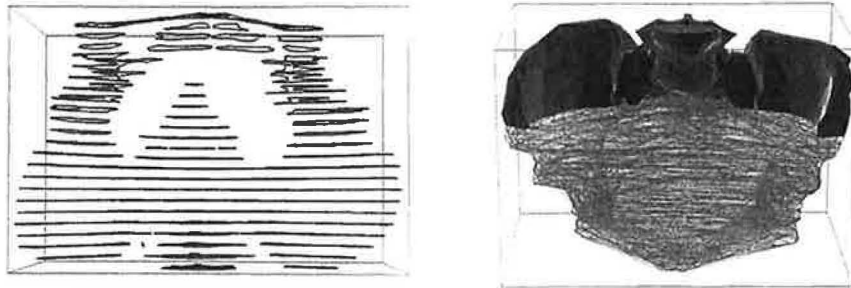


- la *représentation par contours* [30] est un cas très particulier puisqu'elle n'emploie pas de polygones. Elle permet néanmoins de visualiser la surface de l'objet tridimensionnel : les contours donnent une représentation succincte du volume de l'objet lorsqu'ils sont superposés. Cette technique emploie des segments de droites, que l'on

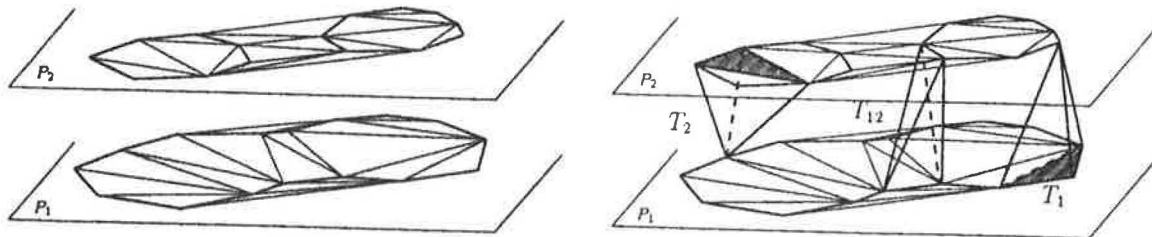
### 3. Tour d'Horizon de la Reconstruction Surfactive.

peut assimiler à des polygones à une dimension, reliant les points de contour de manière séquentielle. L'image de gauche nous montre l'effet obtenu selon deux angles de vue perpendiculaires. Cette technique est bien sûr très adaptée aux objets acquis en coupes (par scanner ou IRM par exemple).

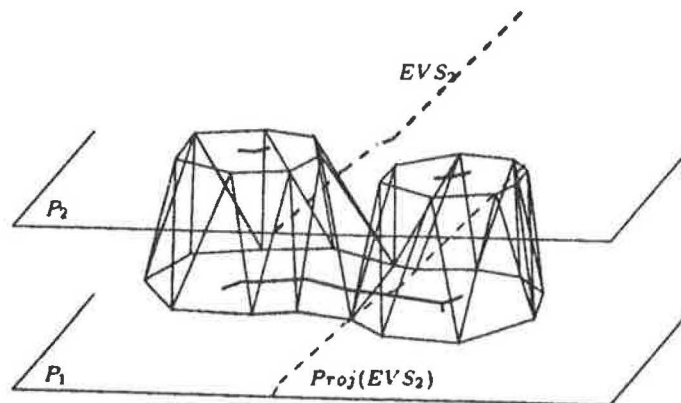
- reconstruction par tétraèdres [31], [32], [33] : elle reprend le principe des segments de droites, mais en l'étendant à la troisième dimension. L'idée principale est de s'appuyer sur les points de contour pour reconstruire les bords, et ce en procédant par couples de coupes voisines. Deux méthodes sont disponibles :



L'algorithme de Boissonnat procède coupe par coupe (à gauche) pour reconstruire la surface polygonale. A droite, la reconstruction de la partie supérieure est terminée, alors que la partie inférieure est en cours de traitement. Cette technique est dite *de couture* et consiste simplement à lier géométriquement les contours.



Une autre approche consiste à réaliser au préalable un partitionnement volumique de l'intérieur des contours (à gauche) en utilisant une *triangulation de Delaunay* [34], [35]. Les coupes sont ensuite mises en correspondance (à droite) en positionnant des tétraèdres dont la base est en  $P_1$  et le sommet en  $P_2$  ou inversement. Cette approche permet d'améliorer très sensiblement la détection de zones ambiguës.



Enfin, les tétraèdres sont filtrés (la crevasse centrale est un cas particulier qui nécessite un traitement approprié par exemple), et seules les faces externes sont conservées : le volume est ainsi recréé. Notons toutefois qu'une telle reconstruction produit une interpolation qui n'est que linéaire entre chaque coupe, ce qui peut être incorrect par rapport à l'objet réel (par exemple l'organe numérisé), mais qui est valable si l'on admet que l'information inter-coupe est de toute façon absente.

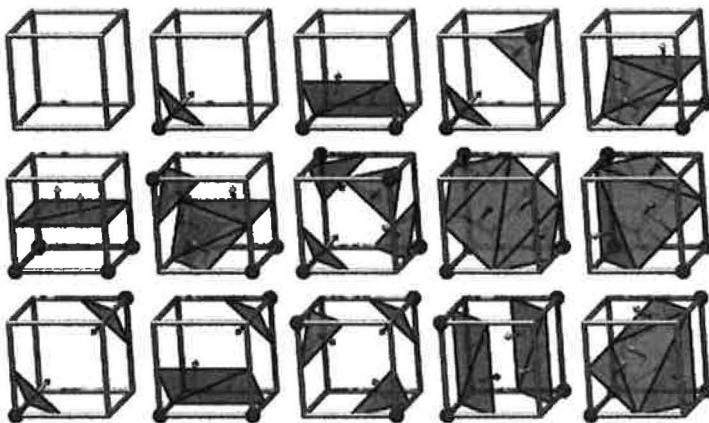


- le *cube opaque* ou *cuberille* [36] : sélectionne les cellules de la grille tridimensionnelle pour lesquelles les valeurs aux sommets encadrent le seuil recherché (une densité par exemple, dans le cas d'un examen médical). Pour chaque cellule ainsi trouvée, six polygones sont générés (un par face de cellule), qui seront ensuite affichés de manière classique. Cette méthode se rapproche de la technique des *marching cubes* qui va suivre.

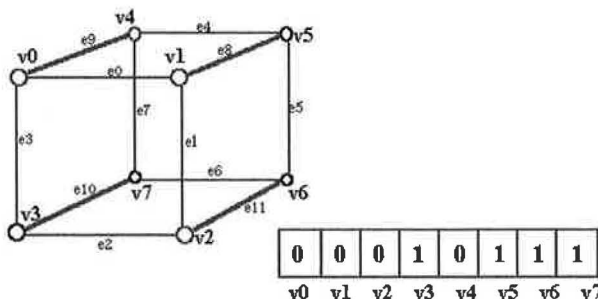


A gauche : image de vertèbre produite par l'algorithme des cubes opaques ou cuberilles. On peut facilement apercevoir les lignes de contour, puisqu'aucun lissage n'est effectué.

- le *cube marcheur* (*marching cube* [37] ou *marching tetrahedra* [38], [39]) : est un raffinement de la méthode des *cuberilles*. C'est un procédé qui ajuste au plus quatre triangles – en utilisant une table de correspondance – dans chaque cellule de la grille tridimensionnelle répondant aux mêmes critères de seuil énoncés pour les cuberilles :



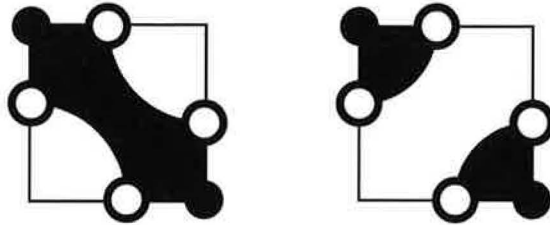
Voici les 15 cas (à gauche) de création de triangles en fonction des valeurs relevées aux sommets de chaque cellule de la grille tridimensionnelle. Il y en a en fait 256, si l'on tient compte de toutes les réflexions et rotations possibles. Les points bleus indiquent les coins où la valeur détectée est supérieure ou égale au seuil recherché.



Les cas sont résolus en utilisant une table de correspondance : les coins sont ordonnés et mis à 1 ou à 0 en fonction de la valeur détectée, ce qui donne sur 8 bits l'index dans la table des configurations polygonales (image précédente).

On remarquera l'existence d'une variante du cube marcheur, nommée *division cubique* (*dividing cube* [40]), qui subdivise les cellules de la grille jusqu'à ce que les polygones générés se projettent en exactement un pixel de l'écran ou qu'un voxel soit atteint. Ce type de technique ne produit donc pas de facettes même s'il n'est pas à proprement parler une méthode de rendu volumique direct.

Il existe néanmoins des cas ambigus quant à la topologie rencontrée dans chaque cellule [41], comme le montre la figure suivante :



Quelle polygonisation adopter ? (en noir : l'intérieur de l'objet)

Ce problème peut être résolu en découpant plus avant la cellule (par subdivision récursive par exemple), ou en introduisant des heuristiques de gradient, d'échantillonnage au centre de la cellule...etc. Bien entendu, cette solution suppose que la définition analytique de l'objet est disponible. Ces palliatifs ne couvrent pas complètement tous les cas possibles. C'est pour cela qu'une extension des cubes marcheurs existe : les *tétraèdres marcheurs* (*marking tetrahedra*) :

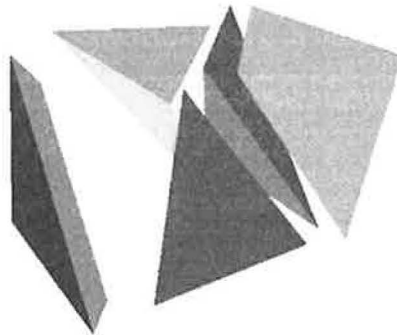
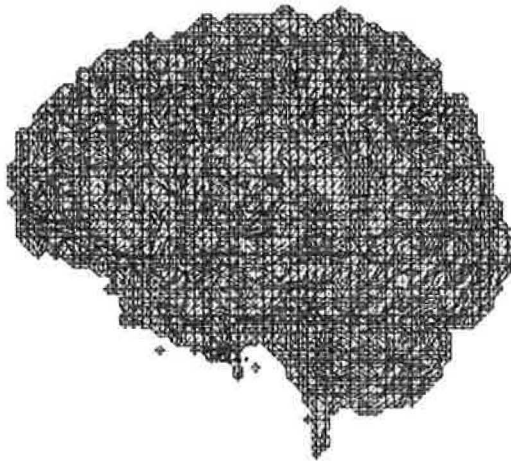


Figure 4

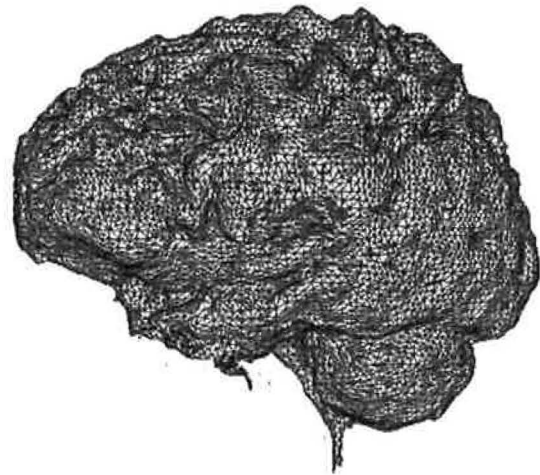
On peut décomposer un cube (une cellule) en 5, 6 ou 24 tétraèdres. On supprime de ce fait certaines ambiguïtés topologiques, mais on augmente le nombre de facettes de manière conséquente. On notera que dans le cas de 5 tétraèdres, les découpages doivent être alternés de manière à faire correspondre les diagonales de cellules adjacentes.

Il s'ensuit une polygonisation dépendant certes du choix du maillage tétraédrique, mais qui donnera toujours un résultat topologiquement cohérent. De plus, la table de correspondance est beaucoup plus réduite : 4 cas en 2D (2 par symétrie), et 5 cas en 3D.



Cette reconstruction (à gauche) a été opérée par *marching cube* : on distingue le motif des cellules, ainsi que le nombre important de polygones. Les contours sont également très carrés selon la méthode employée.

Des techniques de *relaxation* [42], tenant compte des gradients locaux peuvent rendre le maillage plus approprié en déplaçant certains sommets (à droite).



La multitude de polygones utilisés pour représenter les objets tridimensionnels alourdit les traitements opérés dans la chaîne d'affichage et freine les possibilités d'interaction [43]. Pour contrecarrer ce phénomène, des techniques de *simplification surfactive* existent, proposant de diminuer le nombre de facettes tout en préservant au mieux la qualité visuelle et éventuellement topologique [44] de l'objet traité.

### 2.1.2. Représentation analytique du contour : construction indirecte.

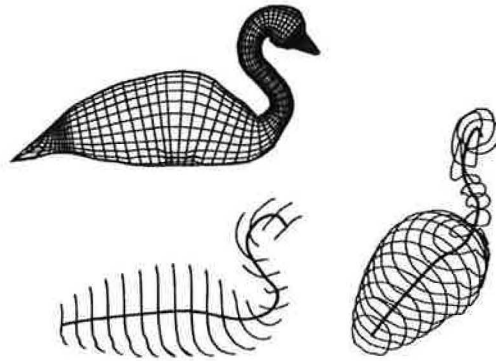


Figure 5

La méthode des cylindres généralisés [4] (Figure 5) construit l'axe principal de l'objet (en bas à gauche) sur lequel sera positionné un cylindre pouvant changer de rayon de manière paramétrique (en bas à droite). L'objet final est représenté en haut à gauche. Des problèmes de robustesse de calcul d'axe ou de détection d'embranchements peuvent survenir. Le formalisme couramment employé est celui des courbes et surfaces paramétrées.



Figure 6

La reconstruction par placement de particules [46] (Figure 6) à l'intérieur du nuage de points est une méthode classique [2][45], qui repose principalement sur le modèle des surfaces implicites. Des contraintes sont nécessaires pour limiter le nombre de primitives ou fournir une caractérisation satisfaisante de l'objet. Des techniques sont développées pour éviter le recours à l'optimisation numérique [47] ou le limiter au maximum [48]. Un cadre générique est présenté pour ce type de reconstruction dans [49].

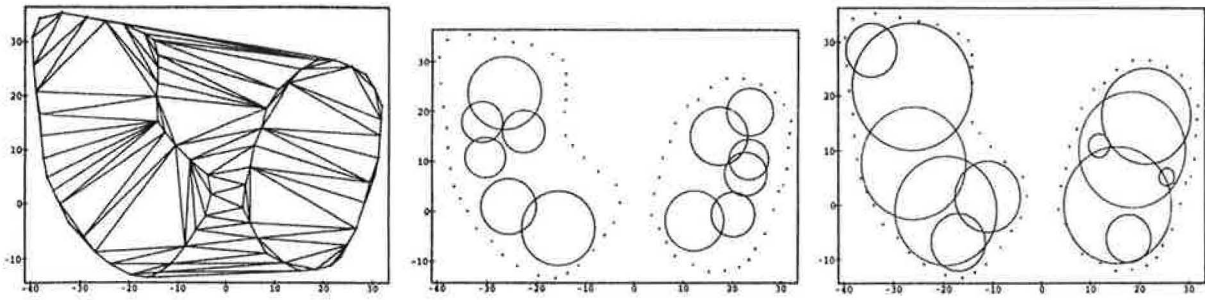
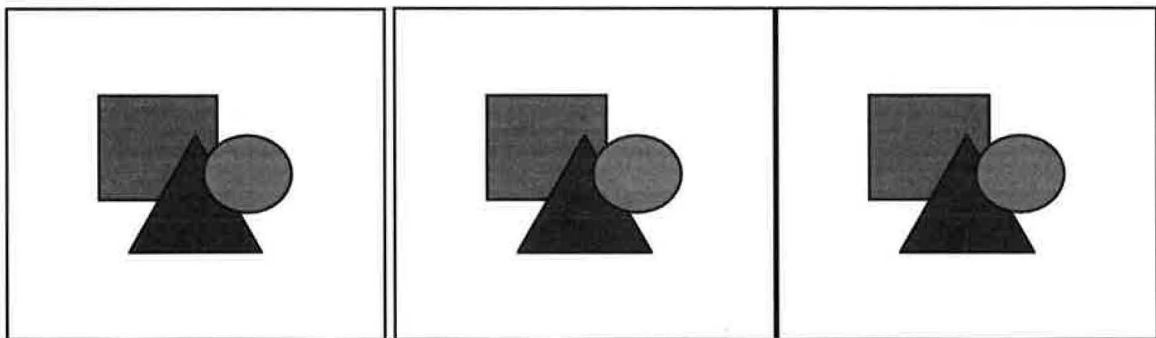


Figure 7

La Figure 7 présente une technique de positionnement de primitives à l'intérieur des contours d'un objet [3] : celle-ci est très complexe et coûteuse en temps de calcul mais illustre bien les difficultés à définir « l'intérieur » d'un objet dont on ne possède que quelques points à la surface.

Nous citerons également deux autres modèles de reconstruction procédurale : les *formes alpha* [25], et les *contours actifs* [27] (ou *snakes*). Ces derniers sont très largement utilisés aussi bien en 2D qu'en 3D, et tout spécialement dans la phase de segmentation. Pour la reconstruction géométrique, le principe s'apparente à celui de la déformation par optimisation numérique [50]



De gauche à droite : le *snake* se déplace vers les contours de l'objet (ici le foie). Il est mû par une énergie relative au gradient rencontré dans l'image qui le guide dans son évolution (Johan Montagnat, Hervé Delingette, INRIA). Cette approche rejoint la reconstruction par optimisation numérique que nous allons développer maintenant.

## 2.2. Reconstruction surfactive par optimisation numérique ou modèle physique.

Nous explorons maintenant la technique de reconstruction surfactive par optimisation numérique, aussi appelée *ajustement surfactive*. Pour de plus amples détails, le lecteur se reportera au chapitre traitant de la reconstruction par E-Squelette (chapitre 6). Le cheminement classique de cette technique est le suivant :

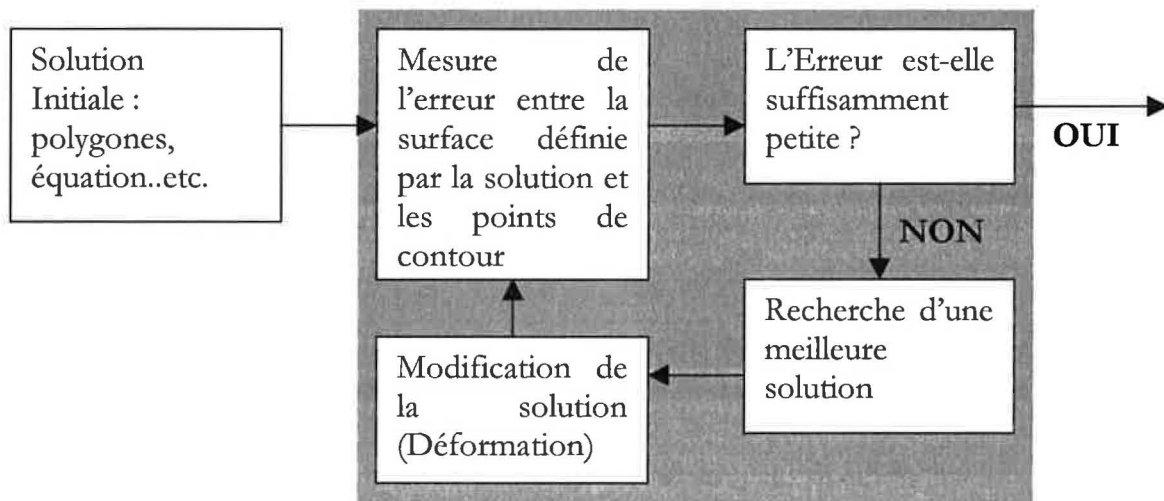


Figure 8

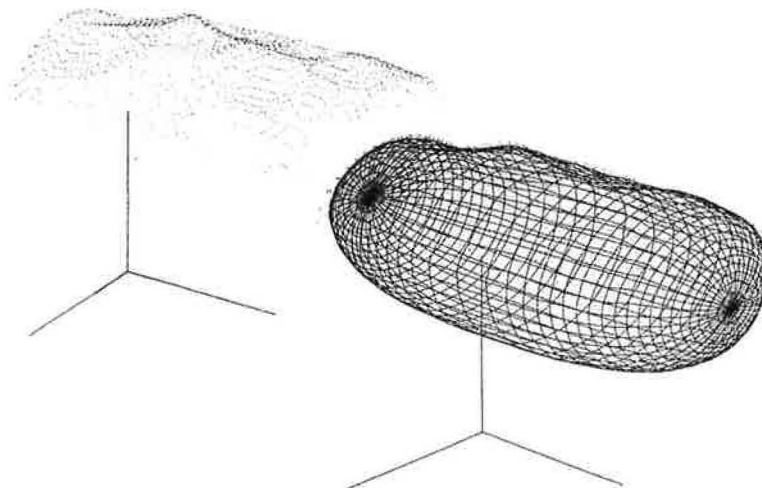


Figure 9

Le principe est simple : un objet est progressivement déformé (à droite) pour approcher au mieux les points de contour (à gauche). Une fonction de déformation adaptée au modèle de représentation est nécessaire [13], [26] : c'est ensuite la technique d'optimisation numérique qui pilote l'ajustement surfactive.

Nous discuterons dans un premier temps des problématiques d'obtention d'une solution initiale puis nous présenterons quelques techniques de recherche appliquées à la reconstruction de surfaces.

### 2.2.1. Obtention de la solution initiale.

Une solution initiale convenable doit satisfaire aux critères suivants :

- avoir une topologie adaptée : l'entité déformable doit posséder les mêmes caractéristiques topologiques que l'objet à reconstruire ;
- être proche des données : bien qu'il soit parfois difficile de caractériser ce critère, il est néanmoins primordial que l'entité déformable soit géométriquement proche des points de contour ;
- être facilement manipulable : ce critère sous-entend que l'entité déformable doit être la plus simple possible afin de limiter les calculs, tels que l'évaluation d'une fonction aux points-clés par exemple.

Une fois celle-ci obtenue par une méthode procédurale comme c'est le cas le plus souvent (voir section 2.1), une méthode d'optimisation numérique prend alors le relais pour la réduire.

### 2.2.2. Les techniques d'optimisation numérique classiques.

Le problème est de minimiser une fonction éventuellement continue et dérivable  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  - typiquement une mesure d'erreur - avec selon le cas des contraintes qui peuvent être :

$$\begin{aligned}g_i(x) &\geq 0, i = 1 \dots m \\ h_j(x) &= 0, j = 1 \dots k\end{aligned}$$

Ces techniques sont très diverses, car chacune répond à des problèmes particuliers. Elles consistent classiquement à déterminer un vecteur  $\mathbf{x}$  de  $\mathbb{R}^n$  qui minimise une fonction  $f(\mathbf{x})$ , où  $f$  n'est pas forcément continue et/ou dérivable. C'est la nature de  $f$  qui conditionne fortement la technique d'optimisation qui devra être plus ou moins robuste - au prix d'une convergence plus lente -, ou s'affranchir des calculs de fonction-dérivée si cette dernière n'est pas disponible.

Ce qui caractérise de manière générale une technique d'optimisation numérique porte sur les aspects suivants :

- le taux de convergence : il peut être *linéaire*, *superlinéaire* ou *quadratique*. Attention cependant puisque ce critère regroupe deux informations. La première est la propriété intrinsèque de l'algorithme, c'est-à-dire le type d'approche qu'il emploie pour parvenir

à la solution. La seconde, plus subtile, dépend directement du problème spécifique qui est posé, comme nous le montre la définition suivante :

Soit  $\{z_i\}_{i=0}^{\infty}$  une séquence dans un *espace de Banach*  $B$  qui converge vers un point  $z^*$ . On dira que  $\{z_i\}$  *converge linéairement* vers  $z^*$  si il existe un entier  $k \geq 0$ , une constante  $E$  et  $\theta \in [0,1)$  tels que :

$$\|z_i - z^*\|_B \leq E\theta^i \quad \forall i \geq k$$

Dans cette définition [6], les éléments  $E$  et  $\theta$  dépendent de la nature du problème.



- la robustesse au *conditionnement* du problème : c'est la sensibilité de la méthode à la manière dont se pose le problème et sa capacité à s'extirper des *extrema locaux*.

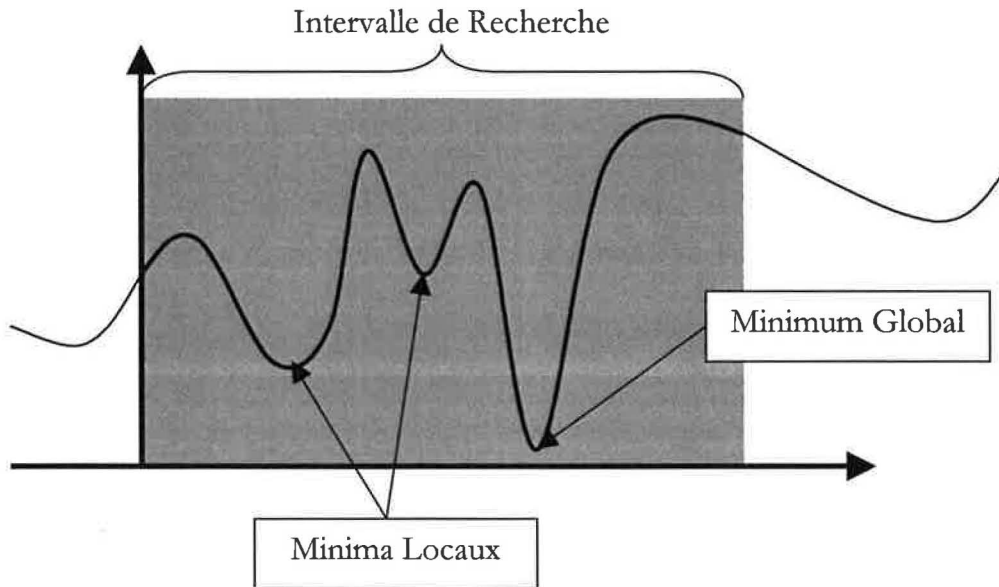


Figure 10

Les méthodes classiques ne permettent sûrement pas d'éviter les minima locaux présentés en Figure 10. Les *métabeuristiques* (voir section 2.2.3) sont par contre très robustes pour ce type de configuration.

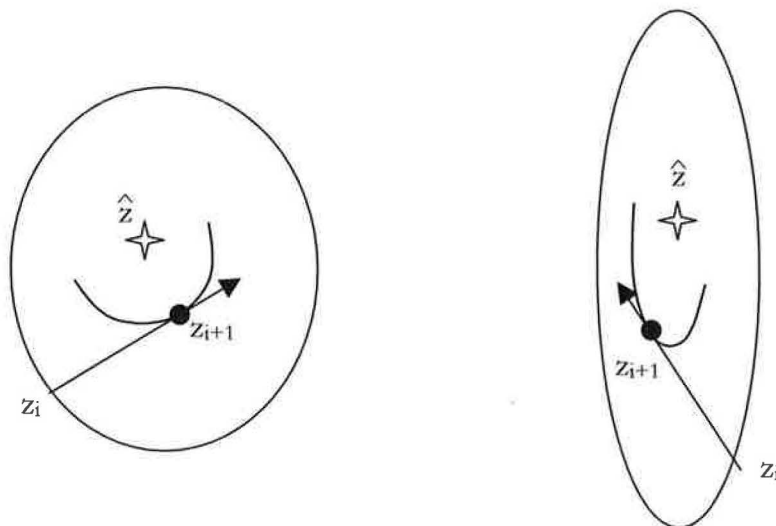


Figure 11

A gauche : problème bien conditionné (*well-conditioned*). A droite : problème mal conditionné (*ill conditioned*) car la vallée est « étroite », ce . augmente les difficultés de définition d'un pas de recherche.

Lorsque le graphe de la fonction  $f$  dont on recherche un minimum  $\hat{z}$  est « étroit », le pas utilisé pour avancer vers la solution doit être très petit. Certaines méthodes comme les

*gradients conjugués* ou *quasi-Newtonienne* [10] permettent de mieux contrôler l'amplitude du pas dans cette situation, à l'inverse des méthodes classiques de recherche par gradient ou de *direction praticable au premier ordre*.

Quelle que soit la solution envisagée, il faut avoir à l'esprit les limitations suivantes :

- certaines méthodes requièrent le calcul de la dérivée de la fonction pour déterminer quelle direction (*gradient*) suivre pour la prochaine solution. Bien que plus puissantes, le coût supplémentaire en temps de calcul peut parfois s'avérer inacceptable ;
- les contraintes appliquées aux variables indépendantes (bornes, valeurs nominales, égalité...) vont conditionner le choix de la méthode ;
- pour  $n$  points, le coût mémoire est soit en  $O(n)$ , soit en  $O(n^2)$  selon la technique utilisée. C'est un paramètre important à prendre en compte puisque  $n$  peut parfois être assez grand ;
- la précision numérique peut être altérée notamment par le calcul des dérivées qui ne sont parfois pas disponibles analytiquement. Sur ce point, certaines méthodes sont plus sensibles que d'autres aux erreurs commises ;
- il est très difficile de prédire le temps que mettra l'algorithme pour rechercher la solution. On peut néanmoins noter trois paramètres importants qui peuvent influencer sur la vitesse de traitement, mis à part la puissance du calculateur sur lequel est développé l'algorithme d'optimisation :
  - le taux de convergence de la méthode (voir plus haut) ;
  - le temps requis pour réaliser une itération, c'est-à-dire pour effectuer un mouvement vers une meilleure solution. Il comprend les évaluations de la fonction  $f$  et de ses dérivées, les inversions de matrices, les calculs de *Jacobien* ou d'*Hessien*...
  - la programmation proprement dite de l'algorithme. Par exemple, relever le seuil en-dessous duquel un petit nombre est considéré comme étant nul peut grandement accélérer le processus, au détriment de la qualité de la solution.

Avant de présenter quelques méthodes d'optimisation numérique, nous concluons en remarquant que le choix de la méthode est loin d'être une science exacte. La famille de méthodes peut dans un premier temps être ciblée en fonction des contraintes à appliquer sur les variables indépendantes, mais il reste alors à sélectionner un des membres de celle-ci. On pourra minimiser le rapport suivant [6] :

$$(\text{indice de conditionnement}) \times (\text{temps par itération})$$

---


$$(\text{taux de convergence})$$

où les valeurs sont définies ainsi :

- indice de conditionnement : il doit caractériser la « forme » de la fonction dans la région de recherche. Une grande valeur sera attribuée à des zones « étroites », une petite valeur à des zones sphériques (ou convexes). On pourra par exemple utiliser le

rapport de la plus grande valeur propre de la matrice Hessienne sur la plus petite calculé au point optimal (en valeur absolue) ;

- taux de convergence : 1 pour la convergence linéaire, 2 pour la convergence quadratique, et entre 1 et 2 pour la convergence superlinéaire.

Nous allons à présent donner quelques indications sur les méthodes existantes, sans toutefois les détailler, de manière à apporter une liste indicative au lecteur. Nous indiquerons les points forts et les points faibles des techniques les plus fréquemment usitées. Certaines des méthodes présentées ici seront développées en détail plus avant dans ce chapitre. De manière générale, le lecteur est invité à se reporter aux ouvrages suivants cités en bibliographie, qui présentent les aspects théoriques mais aussi pratiques des techniques d'optimisation numérique présentés ici : [6], [9], [10], [11], [12].

#### 2.2.2.1. Optimisation classique sans contrainte.

Il s'agit de trouver une valeur minimisant la fonction  $f$ , où  $f$  est continue dérivable.

Méthode de Descente (ou de Montée) Maximale :

- naïve : du fait de son choix le plus simple possible de direction, elle ne peut éviter les extrema locaux.

Méthode de Nelder & Mead (ou *downhill simplex*) :

- ne nécessite aucun calcul de dérivées ( $f$  peut donc être non dérivable) ;
- est coûteuse en terme d'évaluations de  $f$  ;
- ne supporte pas l'optimisation sous contraintes ;
- converge rapidement.

Les Méthodes des Gradients Conjugués et Biconjugués :

- variantes : Klessig-Polak, Polak-Ribière, Fletcher-Reeves ;
- adaptées aux fonctions non-convexes ;
- moins coûteuses en terme d'évaluations que les méthodes Newtonniennes.

#### 2.2.2.2. Optimisation classique avec contraintes d'égalité.

Méthodes de Newton-Raphson et quasi-Newtonniennes :

- adaptées à l'optimisation sous contraintes d'égalité ;
- la fonction doit être continûment dérivable, avec un Jacobien non-singulier ;

- possèdent de nombreuses variantes, agissant notamment sur le taux de convergence et améliorant certains comportements : Goodman-Lance, McGill-Kenneth, Bellman-Kalaba, Abramov...

### 2.2.2.3. Optimisation classique avec contraintes d'égalité et d'inégalité.

Méthode de Fonction de Pénalité.

Méthode des Directions Praticables de Second Ordre :

- deux variantes : recherche linéaire et recherche quadratique ;
- $f$  doit être deux fois continûment dérivable.

Méthode des Centres.

Méthode de Projection de Gradient.

### 2.2.3. Les Métaheuristiques.

Ces techniques [7] sont une alternative pour l'optimisation. Le terme *métaheuristique* traduit le fondement générique de celles-ci, et leur adaptabilité à une large classe de problèmes.

La Figure 12 présente une taxonomie des métaheuristiques :

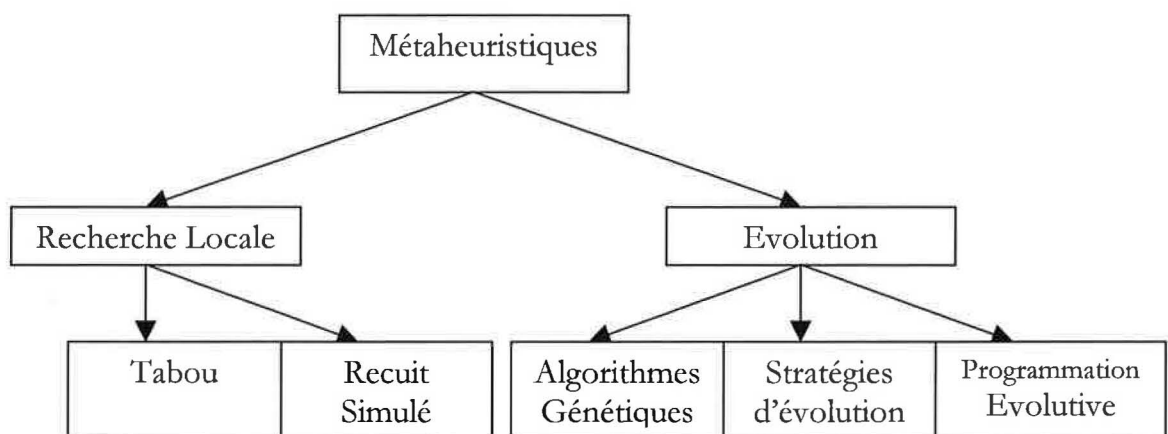


Figure 12

Des méthodes hybrides combinant plusieurs approches existent également. Les méthodes d'évolution étant employées en Intelligence Artificielle, elles ne seront pas abordées dans ce chapitre. Ce sont les techniques de recherche locales qui nous intéressent plus

particulièrement : nous les survolerons dans cette section, et nous développerons complètement l'algorithme *tabou* [21] dans le chapitre consacré à la reconstruction surfactive par E-Squelette.

#### 2.2.3.1. La Méthode Tabou.

Cette technique sera entièrement développée dans le chapitre traitant de la reconstruction surfactive par E-Squelette.

#### 2.2.3.2. La Méthode du Recuit Simulé.

Elle s'inspire du processus de recuit physique employé en métallurgie. Cette technique vise à améliorer la qualité d'un matériau en cherchant un état d'énergie minimale correspondant à une structure stable du matériau. En partant d'une haute température à laquelle la matière est devenue liquide, la phase de refroidissement progressif conduit la matière à retrouver sa forme solide. Chaque température est maintenue jusqu'à ce que la matière trouve un équilibre thermodynamique. Quand la température tend vers zéro, seules les transitions vers un état d'énergie plus faible sont possibles.

Le Recuit Simulé date des années 50 : Metropolis et al. [14] utilisent une méthode stochastique pour générer une suite d'états successifs du système en partant d'un état initial donné. Tout nouvel état est obtenu en faisant subir une perturbation aléatoire à un atome quelconque. Soit  $\Delta E$  la différence d'énergie occasionnée par une telle perturbation. Le nouvel état est accepté si l'énergie du système diminue (typiquement la fonction d'erreur) i.e.  $\Delta E \leq 0$ , sinon il est accepté avec la probabilité suivante :

$$p(\Delta E, T) = e^{\left(\frac{\Delta E}{C_B \cdot T}\right)}$$

Équation 1

où  $T$  est la température du système et  $C_B$  une constante physique connue sous le nom de *constante de Boltzmann*.

Après un grand nombre de perturbations, un tel processus fait évoluer le système vers un état d'équilibre thermodynamique selon la *distribution de Boltzmann* qui est définie par la probabilité de se trouver dans un état  $E$  :

$$\text{Pr}(\Delta E, T) = c(T) \cdot e^{\left(\frac{\Delta E}{C_B \cdot T}\right)}$$

Équation 2

où  $c(T)$  est un facteur de normalisation.

Le recuit simulé est donc une méthode de recherche locale, qui cherche des configurations de coût plus faible tout en acceptant de manière contrôlée des configurations qui dégradent la fonction d'erreur. A chaque nouvelle itération, un voisin de la solution courante est généré de manière aléatoire, et selon le cas il est soit retenu soit rejeté selon les règles dictées précédemment.

Cette méthode est principalement régie par deux paramètres :

- la longueur des paliers de température ;
- la fonction calculant la suite décroissante (mais pas forcément strictement) des températures.

Il existe de nombreuses heuristiques concernant ces paramètres [15], [16], [17], souvent liées aux applications aujourd'hui très vastes [18], [19], [20]. La preuve de convergence a également beaucoup contribué à sa popularité.

### 3. CONCLUSION.

La diversité des méthodes de reconstruction et de modèles met à disposition de vastes possibilités de représentation des données tridimensionnelles. Pourtant chaque solution s'accompagne de limitations particulières qui font que, *in fine*, c'est la nature de l'objet numérisé qui guidera les décisions. Les objets de nature organique, constitués de formes douces, seront par exemple mieux représentés par des fonctions implicites car elles autorisent des soudures lisses et continues faciles à mettre en œuvre. La topologie conditionnera la technique d'optimisation à employer pour améliorer la représentation initiale au sens d'un critère d'erreur : il est en effet évident que les objets convexes ou fortement concaves ne poseront pas les mêmes difficultés en ce qui concerne l'obtention d'une solution initiale et la recherche d'une bonne approximation surfactive.

Les modèles déformables présentent une grande résistance au bruit ainsi qu'une bonne extrapolation des données absentes (l'espace entre les points de capture par exemple). Ceci est dû aux contraintes de régularisation appliquées ou inhérentes à ce type d'entité. *A contrario*, ils restent très sensibles aux données aberrantes résultant d'erreurs algorithmiques en acquisition (*outliers*), et sont très sujets aux problèmes d'extrema locaux. Des traitements existent dans certains cas pour remédier à ce type de situation, nous l'expliquerons dans le chapitre 6 au travers de la présentation de notre méthode de reconstruction.

#### 4. BIBLIOGRAPHIE.

- [1] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, « *Surface Reconstruction from Unorganized Points* », Computer Graphics, 26 (2), pp. 71-78, July 1992
- [2] S. Pontier, B. Shariat and D. Vandorpe, « *Shape Reconstruction using Skeleton-Based Implicit Surface* », proceedings of 9<sup>th</sup> Canadian Conference on Computational Geometry, pp. 205-210, 1998
- [3] C. T. Lim, G. M. Turkiyyah, M. A. Ganter and D. W. Storti, « *Implicit Reconstruction of Solids from Cloud Point Sets* », Solid Modeling'95, pp. 393-402, ACM, 1995
- [4] F. Lazarus, A. Verroust, « *Décomposition Cylindrique de Polyèdre et Courbe Squelette* », proceedings of 3<sup>èmes</sup> Journées AFIG, pp. 255-264, Novembre 1995
- [5] D. Terzopoulos, « *Dynamic 3D Models with Local and Global Deformations : Deformable Superquadrics* », IEEE Transactions on PAMI, 13(7), pp. 703-714, July 1991
- [6] E. Polak, « *Computational Methods in Optimization : A Unified Approach* », Mathematics in Science and Engineering, vol. 77, Edited by Richard Bellman, Academic Press, 1971
- [7] J-K. Hao, P. Galinier and M. Habib, « *Métaheuristique pour l'Optimisation Combinatoire et l'Affectation sous Contraintes* », Rapport de Synthèse LGI2P EMA-ERIEE/LIRMM URA CNRS, [hao@eerie.fr](mailto:hao@eerie.fr)
- [8] P. Bézier, « *Courbes et Surfaces* », Mathématiques et CAO, vol. 4, Hermès Publishing, 1986
- [9] J. Cartier, G. Laporte, R. Parent and J-M. Picard, « *Optimisation* », Méthodes Quantitatives, vol. 8, Editions Sciences et Cultures Inc., Montreal, 1977
- [10] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, « *Numerical Recipes in C : The Art of Scientific Computing Second Edition* », Cambridge University Press, 1988-1992
- [11] J. Verschelde, « *Homotopy Continuation Methods for Solving Polynomials Systems* », PhD. Thesis, U.D.C. 681.3 F21, Katholieke Universiteit Leuven, May 1996
- [12] P. Pelletier, « *Techniques Numériques appliquées au Calcul Scientifique* », Masson et Cie. Editeurs, 1971
- [13] A. H. Barr, « *Global and Local Deformations of Solid Primitives* », Computer Graphics, 18(3), pp.21-30, July 1984
- [14] W. Metropolis, A. Roenbluth, M. Rosenbluth, A. Teller, E. Teller, « *Equation of the State Calculations by Fast Computing Machines* », Journal of Chemical Physics, 21, pp. 1087-1092, 1953
- [15] S. Lundy, A. Mees, « *Convergence of an Annealing Algorithm* », Mathematical Programming, 34, pp. 111-124, 1986
- [16] G. Ducek, T. Scheuer, « *Threshold Accepting: a General Purpose Optimization Algorithm* », Journal of Computational Physics, 90, pp. 161-175, 1990
- [17] D. Connoly, « *An Improved Annealing Scheme for the QAP* », European Journal of Operational Research, 46, pp. 93-100, 1990
- [18] E. H. L. Aarts, J. K. Lenstra, « *Local Search in Combinatorial Optimization* », John Wiley & Sons, 1997



- [19] C. Koulamas, S. R. Anthony and R. Jean, « *A Survey of Simulated Annealing Application to Operations Research Problem* », Omega, 22, pp. 41-56, 1994
- [20] R. V. Vidal, « *Applied Simulated Annealing* », Lecture Notes in Economics and Mathematical Systems, 396, Springer-Verlag, Berlin, 1993
- [21] F. Glover, « *Future Paths for Integer Programming and Links to Artificial Intelligence* », Computers and Operations Research, 13, pp. 533-549, 1986
- [22] P. Hansen, « *The Steepest Ascent Mildest Descent Heuristic for Combinatorial Programming* », Congress on Numerical Methods in Combinatorial Optimization, Capri, Italie, 1986
- [23] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce and D. J. Kriegman, « *Parameterized Families of Polynomials for Bounded Algebraic Curve and Surface Fitting* », IEEE Transactions on PAMI, 16(3), pp. 287-303, March 1994
- [24] D. R. Forsey, R. H. Bartels, « *Hierarchical B-spline Refinement* », Computer Graphics, 22(3), pp. 205-212, Proceedings of SIGGRAPH88, August 1988
- [25] B. Guo, J. Menon and B. Willette, « *Surface Reconstruction Using Alpha Shapes* », Proceedings of Eurographics'96, 1996
- [26] S. Sclaroff, A. Pentland, « *Generalized Implicit Functions for Computer Graphics* », Computer Graphics, 25(4), pp. 247-250, July 1991
- [27] M. Kass, A. P. Witkin, and D. Terzopoulos, « *Snakes: Active Contour Models* », IJCV, No. 4, pp. 321-331, January 1988
- [28] J. Bloomenthal, « *Surface Tiling* », Implicit Surfaces, Chapter 4
- [29] P. Ning, J. Bloomenthal, « *An Evaluation of Implicit Surface Tilers* »
- [30] P. Borianne, M. Jaeger, « *Polygonisation Réversible en Imagerie Médicale. Application à la Visualisation de Divers Tissus Anatomiques* », Actes CARI'94
- [31] J-D. Boissonnat, « *Shape Reconstruction from Planar Cross Sections* », Computer Vision, Graphics and Image Processing, 44, pp. 1-29, Academic Press, 1988
- [32] J-D. Boissonnat, B. Geiger, « *Three Dimensional Reconstruction of Complex Shapes Based on the Delaunay Triangulation* », RR N°1697, INRIA, Mai 1992
- [33] H. Fuchs, Z. M. Kedem and S. P. Uelson, « *Optimal Surface Reconstruction from Planar Contours* », pp. 693-702, Communications of the ACM, Volume 20, Number 10, October 1977
- [34] P. P. Watson, « *Computing the n-dimensional Delaunay Triangulation with Application to Voronoi Polytopes* », pp. 167-172, Computer Journal, Volume 24, Number 2, 1981
- [35] H. Edelsbrunner and R. Seidel, « *Voronoi diagrams and arrangements* », pp. 25-44, Discrete Comput. Geom. 1, 1986.
- [36] G. T. Herman, H. K. Liu, « *Three Dimensional Display of Human Organs from Computed Tomograms* », pp. 1-21, Computer Graphics and Image Processing, Volume 9, Number 1, January 1979
- [37] W. E. Lorensen, H. E. Cline, « *Marching Cubes : A High Resolution 3D Surface Construction Algorithm* », pp. 163-169, Computer Graphics, Volume 21, Number 4, July 1987
- [38] G. M. Treece, R. W. Prager and A. H. Gee, « *Regularised Marching Tetrahedra : Improved Iso-Surface Extraction* », CUED/F-INFENG/TR 333, Cambridge University Engineering Department, September 1998

- [39] P. Shirley, A. Tuckman, « A Polygonal Approximation to Direct Scalar Volume Rendering », pp. 63-70, Computer Graphics, Volume 24, Number 5, November 1990
- [40] H.E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford and B.C. Teeter, « Two Algorithms for the Three Dimensional Reconstruction of Tomograms », pp. 320-327, Med. Phys., 15 (3), Am. Assoc. Phys. Med, May/June 1988
- [41] M. J. Dürst, « Letters : Additional Reference to 'Marching Cubes' », pp. 72-73, ACM Computer Graphics, Volume 22, Number 4, 1988
- [42] K. Shimada, D. C. Gossard, « Bubble Mesh : Automated Triangular Meshing of Non-Manifold Geometry by Sphere Packing », pp. 409-419, ACM Proceedings of Solid Modeling'95, 1995
- [43] W. Seibold, G. Wyll, « Two many triangles. » , p. 242, Computer Graphics, SIGGRAPH'98, Conference abstracts and applications, 1998
- [44] T. Stander, J. C. Hart, « Guaranteeing the Topology of an Implicit Surface Polygonization », Computer Graphics, 1997
- [45] S. Muraki, « *Volumetric Shape Description of Range Data using « Blobby Model»* », Computer Graphics, SIGGRAPH91, 25(4), pp. 227-235, July 1991
- [46] E. Bittar, N. Tsingos and M-P Gascuel, « *Automatic Reconstruction of Unstructured 3D Data : Combining a Medial Axis and Implicit Surfaces.* », Proceedings of Eurographics'95, september 1995
- [47] E. Ferley, M-P. Cani-Gascuel and D. Attali, « *Skeletal Reconstruction of Branching Shapes.* », Computer Graphics Forum, Volume 16 (15), december 1997
- [48] N. Tsingos, E. Bittar and M-P. Gascuel, « *Semi-automatic Reconstruction of Implicit Surfaces for Medical Applications.* », Computer Graphics International'95, Leeds, UK, June 1995
- [49] C.Guiard. « *Modèles de surfaces déformables définis et contraints par un ensemble d'informations structurelles.* », PhD thesis, Université de la Méditerranée, Marseille, 2000.
- [50] H. Delingette, « *General Object Reconstruction based on Simplex Meshes.* », Rapport de recherche N°3111, Thème 3, INRIA, Février 1997



## **4. Tour d'Horizon de la Reconnaissance de Formes.**

« Je soupçonne cependant qu'il n'était pas très capable de penser.  
Penser, c'est oublier des différences, c'est généraliser, abstraire.  
Dans le monde surchargé de Funès il n'y avait que des détails, presque immédiats. »  
Jorge Luis Borges, Funès ou la Mémoire, Fictions

## 1. INTRODUCTION.

D'une manière générale, la théorie de la *reconnaissance des formes* s'attache à expliquer et reproduire la manière dont nous extrayons l'information à partir de stimuli complexes. Le terme *forme* désigne ici la composition d'un ensemble de stimuli que l'observateur associera à une classe d'objets connus. Il recouvre donc un sens bien plus vaste que celui désignant la forme géométrique d'un objet, ce qui traduit bien la nécessité d'appréhender le processus de reconnaissance au sens large. Les objets connus auront quant à eux été stockés en mémoire au cours d'expériences passées. Pour la *reconnaissance automatique*, c'est-à-dire la classification d'objets par un calculateur, il est courant de distinguer les problématiques 2D et 3D. Cette différenciation vient principalement du fait que les acquisitions ne sont alors plus réalisées de la même manière, ne fournissant que des données partielles ou sujettes à ambiguïtés. Néanmoins nous ne nous étendrons pas sur les moyens de perception : nous développerons plutôt les techniques de reconnaissance fondamentales, toujours sous-jacentes aux méthodes actuelles.

Une chose semble acquise : la reconnaissance automatique n'a pas, durant ces vingt dernières années, apporté le succès escompté :

« *But a robot that can clear the table, throw the eggshells in with the garbage and wash up the dishes, instead of washing up the eggshells and throwing the dishes in the garbage, is still some distance off.* »

Qui donne à peu près ceci :

« *Le jour où un robot jettera les coquilles d'œuf à la poubelle et lavera la vaisselle au lieu de laver les coquilles d'œuf et jeter la vaisselle à la poubelle est encore assez loin.* » [25]

Equiper aujourd'hui un robot de capteurs de vision est très simple et économique. Le problème se situe en aval : quels algorithmes utiliser pour reconnaître ce que voit le robot ? Ce que maîtrise de manière tout à fait ordinaire un jeune enfant semble très difficile à formaliser et mettre en place dans le contexte de la vision artificielle.



Figure 1

Comment détecter ces deux baleines, lorsque la seule information est un ensemble de points colorés ? C'est le rôle du traitement d'image, qui doit isoler les entités « pertinentes » et les identifier comme objets uniques. On notera les erreurs de contour dans l'image de droite, preuve qu'un tel travail n'est pas aussi simple qu'il y paraît. Dans ce chapitre nous nous attacherons aux méthodes qui suivent ce type de traitement et qui visent à la classification des entités observées.

Dans le processus de reconnaissance, deux cheminements sont possibles :

- le traitement est dicté par les données : l'algorithme de reconnaissance est influencé par la nature des informations en entrée ;
- le traitement est dicté par le concept : des hypothèses sur la nature de l'information ainsi que son contenu sont formulées en fonction du contexte.

Des combinaisons multiples de ces deux cheminements sont employées, donnant naissance à plusieurs approches que nous allons découvrir.

## 2. APPROCHES POUR LA RECONNAISSANCE DE FORMES.

Les sections 2.1 et 2.2 sont des cas particuliers de la technique présentée en 2.3. Quant à celle présentée en 2.4, elle propose un cheminement totalement différent.

### 2.1. Reconnaissance par motif.

Cette technique consiste à faire correspondre un motif stocké en mémoire avec les données sensorielles en entrée. Elle s'inspire du fait que nous stockons en mémoire à long terme les connaissances que nous avons acquises dans les expériences passées.

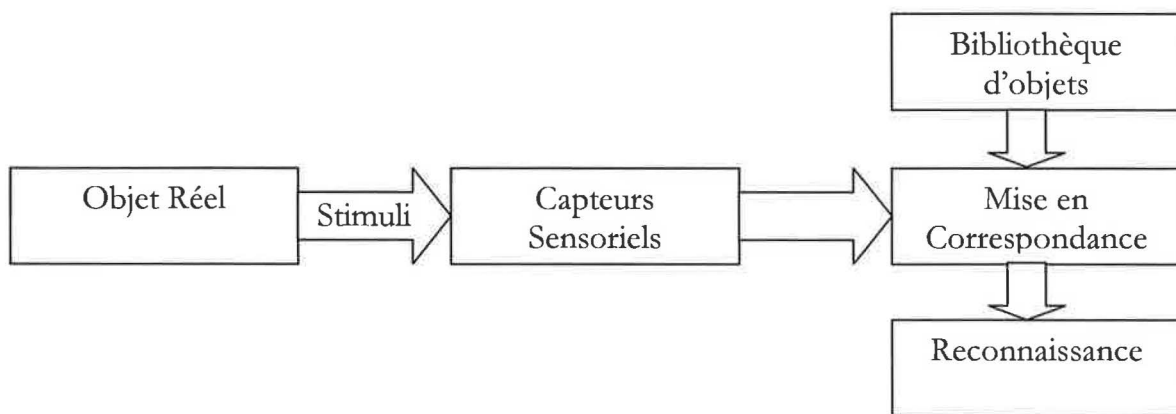


Figure 2

La Reconnaissance par Motif

Une bibliothèque d'objets connus est parcourue, et une mise en correspondance est réalisée avec les données d'entrée jusqu'à obtenir un résultat : le motif est « proche » de l'objet, ou aucun élément de la bibliothèque n'est valable.

La théorie de reconnaissance par motif est purement dictée par les données et les motifs présents dans la bibliothèque : la reconnaissance dépend de ces deux types d'information.

Les avantages de cette technique sont les suivants :

- elle est facilement parallélisable, c'est-à-dire que plusieurs motifs peuvent être testés en même temps sur les données d'entrée ;
- elle n'est soumise à aucune hypothèse de contexte, et peut donc s'avérer efficace pour des détections inattendues d'objets ;
- elle se rapproche de certains aspects de la reconnaissance humaine, comme par exemple la rotation mentale.

On dénombre toutefois plusieurs problèmes:

- chaque objet nécessite un motif correspondant dans la bibliothèque, y compris toutes les variations d'un même objet (différents modèles de table par exemple) ;
- pour être reconnu, chaque objet doit subir un recalage (orientation, taille ...etc), ce qui induit une connaissance des modalités d'acquisition (par exemple : les paramètres d'une caméra) ;
- l'algorithme est sensible aux distorsions (objet occulté partiellement ou déformé durant l'acquisition).

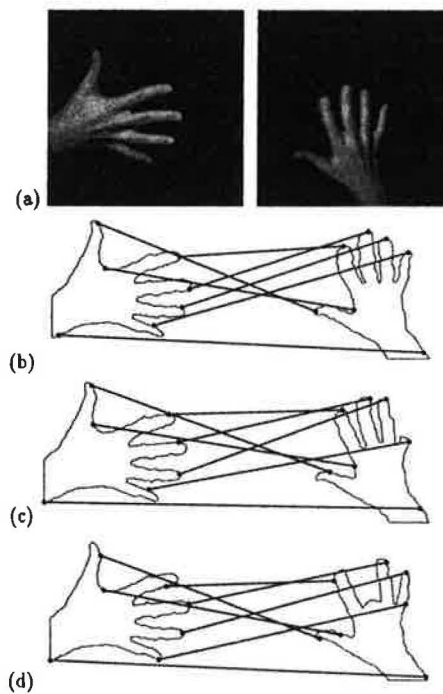


Figure 3

La reconnaissance par motif : la main (à gauche) est un objet de référence contenu dans la bibliothèque [20]. La main à droite représente les données sensorielles d'entrée : par un jeu de points de correspondance, le motif est aligné avec l'objet observé, et la main est détectée. Les figures b, c et d présentent trois acquisitions différentes, dans lesquelles l'ordinateur retrouve les correspondances malgré le masquage de certains doigts.



## 2.2. Reconnaissance par parties pertinentes.

Elle consiste à décomposer les motifs en parties présentant un intérêt particulier (géométrie, couleur...). Ce sont ces éléments qui seront ensuite stockés dans une bibliothèque de référence [24].

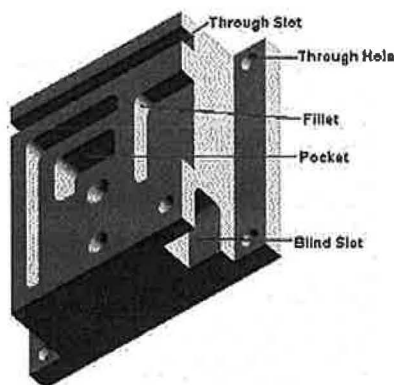


Figure 4

Voici un exemple de représentation par parties pertinentes : au lieu de stocker exactement la géométrie de l'objet, on stocke les « spécificités » de celui-ci comme les trous, chanfrein, etc.

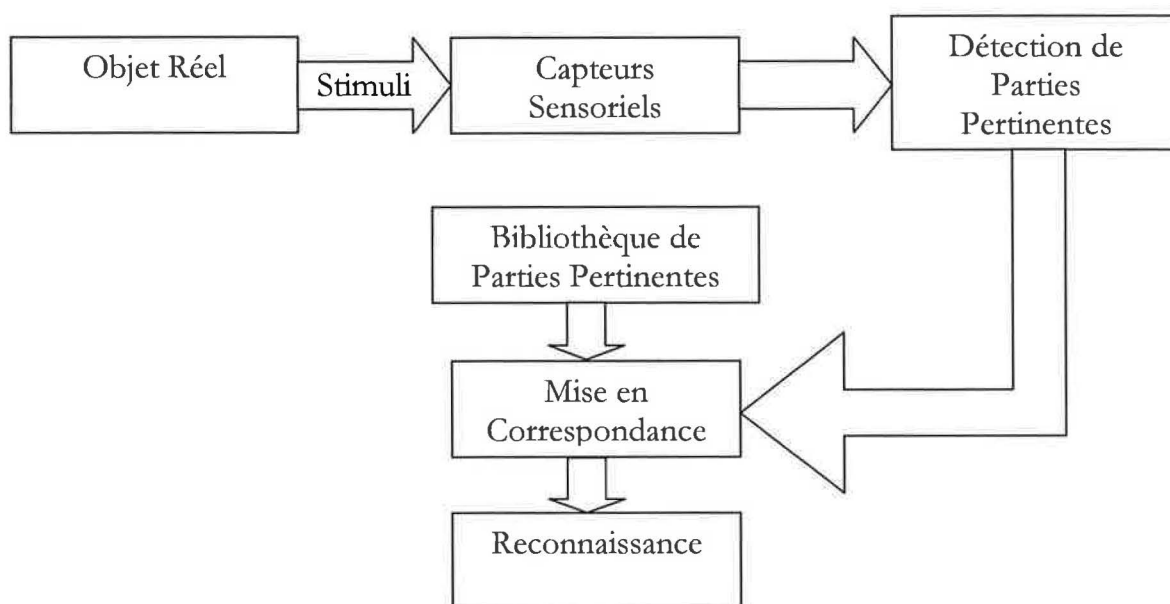


Figure 5

La Reconnaissance par Parties Pertinentes : remarquons que la phase de mise en correspondance des parties pertinentes utilise le principe de la reconnaissance par motif.

Avantages :

- la représentation est beaucoup moins sensible aux variations en entrée ainsi qu'au bruit ;
- la détection peut réussir même s'il manque certaines parties pertinentes ;
- le stockage des parties pertinentes est plus économique que celui des motifs complets ;
- des études ont montré que même dans le cerveau humain des neurones spécialisés s'activaient individuellement lorsque des objets présentant certains caractères étaient exhibés. C'est le cas par exemple pour les lignes horizontales et verticales qui stimulent séparément des zones spécifiques du cortex visuel.

Inconvénients :

- il est parfois difficile de définir ce qu'est une « partie pertinente » sur un objet ;
- si le dictionnaire des parties pertinents possibles est insuffisant, certains objets ne pourront pas être classés ou ne seront pas caractérisés convenablement ;
- une recherche en série par parties pertinentes peut être coûteuse en temps de calcul.

Cette technique peut combiner les cheminements : la mise en correspondance des parties pertinentes est dicté par les données (comme avec la reconnaissance par motif), alors que certaines parties peuvent être tacitement attendues avant même d'avoir été détectées, ce qui induit une approche conceptuelle. De cette manière il devient possible d'accélérer le processus de recherche dans la bibliothèque.

### **2.3. Reconnaissance structurelle.**

Cette approche [5][38], généralise les techniques précédentes, en mettant l'accent sur l'organisation spatiale des composants élémentaires (ou parties pertinentes). La théorie des *geons* ou (*Geometric Ions*) [11] suppose que nous décomposons les objets en primitives géométriques tridimensionnelles (cônes, cylindres...), disposées dans l'espace, qui nous permettent ensuite l'identification en détectant les sous-parties correspondantes.



Figure 6

La Reconnaissance Structurelle : les parties pertinentes sont extraites et reconnues mais également leur arrangement spatial. Dans le cas présenté ici, toute autre arrangement invaliderait la reconnaissance d'un visage.

Avantages supplémentaires par rapport à l'approche par parties pertinentes :

- la configuration spatiale est prise en compte, ce qui renforce les succès d'identification ;
- nous sommes très sensibles aux arrangement particuliers des parties d'un objet (lignes horizontales, verticales, parallèles...), ce qui semble être en accord avec cette approche. Par exemple, il est plus facile pour l'être humain de reconnaître des photos d'un visage quand il est à l'endroit, preuve d'une prédisposition particulière aux stimuli d'entrée.

Inconvénients :

- identiques à la reconnaissance par parties pertinentes.

Remarquons que ce type de reconnaissance est comparable en théorie des graphes [2] [3] à la mise en correspondance d'arbres ou de graphes dont les nœuds sont valués : des contraintes de minimisation de différence sur les valeurs internes des nœuds ainsi que de non violation de la topologie sont appliquées pour guider l'appariement. Certaines méthodes utilisent par ailleurs la formulation en graphe pour reconnaître ou comparer des objets [1] [6] [4].

## **2.4. La reconnaissance statistique.**

Elle [36][37] regroupe les techniques qui nécessitent un ensemble d'apprentissage, c'est-à-dire un ensemble de formes dont la classe d'appartenance est connue . En visualisant cet ensemble, l'algorithme réduit la dimension de l'espace de représentation pour se focaliser sur les mesures les plus pertinentes. Le réseaux de neurones, que nous découvrirons dans ce chapitre, est un cas typique de la reconnaissance statistique.

### **3. LA PRISE DE DÉCISION OU COMMENT MESURER LA CORRESPONDANCE.**

Nous présentons les fondamentaux des techniques de reconnaissances structurales et statistiques. La partie consacrée aux réseaux de neurones (3.4) est propre à la reconnaissance statistique, mais toutes les autres sections sont plus générales et expriment les manières de prendre une décision à partir de données complètes ou incomplètes.

#### **3.1. Qu'est-ce qu'un attribut ?**

Un attribut est un paramètre mesuré sur l'objet détecté : ce peut être une longueur, une couleur, un volume ...Etc. C'est ce paramètre qui servira à décider de l'appartenance de l'objet à une classe donnée – selon les approches indiquées au paragraphe précédent.

Un attribut peut être :

- linéaire, dans ce cas il est soit continu, soit discret ;
- nominal (ou symbolique), il prend donc des valeurs discrètes mais non nécessairement ordonnées linéairement. Un ensemble de couleurs par exemple {vert, rouge bleu, blanc} peut certes être représenté par l'ensemble {1, 2, 3, 4}, mais employer des distances classiques sur ce type de paramètre n'est pas pertinent.

Le problème classique est de déterminer les attributs les meilleurs possibles, c'est-à-dire [20] :

- qu'ils soient invariants au sein d'une même classe, en d'autres termes qu'ils caractérisent au mieux un type d'objet particulier ;
- qu'ils soient robustes aux conditions d'acquisition : le bruit, la distorsion, le sous-échantillonnage ainsi que la variabilité naturelle (par exemple différence d'écriture entre deux personnes) sont autant de paramètres qui agissent sur la valeur des attributs.

#### **3.2. La mesure de distance.**

C'est l'outil principal sur lequel vont s'appuyer les méthodes décisionnelles de reconnaissances [21]. Les mesures de distance peuvent bien sûr être utilisées en tant que telles, c'est pourquoi elles figurent dans cette section.

Un grand nombre de fonctions de distance a été proposé, pour décider de l'appartenance d'un objet à une classe. Ces métriques s'appliquent avec succès aux données numériques, mais sont difficiles à exploiter sur des mesures discrètes ou non ordonnées (on dit encore *nominales*).

Le choix d'une fonction de distance influence directement le *biais* d'un algorithme de reconnaissance, c'est-à-dire qu'il conditionne les résultats généraux en sortie [26]. Il a été

démontré qu'aucun algorithme d'apprentissage ne pouvait donner un résultat général meilleur qu'un autre si on l'étend à tous les problèmes possibles [27]. Aucune fonction de distance n'est ainsi meilleure qu'une autre en termes de généralisation, mais certaines s'adaptent plus à des classes particulières de problèmes [29].

La normalisation est souvent requise lorsqu'on utilise une fonction de distance linéaire comme la distance euclidienne par exemple. De cette manière, aucun paramètre ne prend plus d'importance qu'un autre dans la mesure. Pour ce faire, on divise usuellement par la valeur maximale que peut prendre une valeur ou par la déviation standard [28]. On peut encore décider d'ôter les 5% de valeurs minimales et maximales, afin de s'affranchir d'éventuelles erreurs dans un ensemble de mesures.

La pondération peut également être introduite dans la fonction de distance [30] selon le domaine d'application, même si cela s'avère parfois très difficile à mettre en place. Les critères définissant les pondérations sont très dépendants du domaine d'application et requièrent de ce fait la présence d'experts.

Nous allons présenter quelques fonctions de distance afin de préciser leur rôle dans l'étape de mesure. Nous définissons dans la suite de ce chapitre  $\mathbf{x}$  et  $\mathbf{y}$  comme étant deux vecteurs de dimension  $m$ .

La distance la plus classique est la distance euclidienne :

$$E(\mathbf{x}, \mathbf{y}) \stackrel{def}{=} \sqrt{\sum_{i=1}^m (x_i - y_i)^2}$$

Équation 1

Distance de Manhattan, qui mesure la distance entre deux points selon des axes orthogonaux (très employée en électronique où les chemins sont souvent parallèles aux axes  $x$  et  $y$ ) :

$$D(\mathbf{x}, \mathbf{y}) \stackrel{def}{=} \sum_{i=1}^m |x_i - y_i|$$

Équation 2

Distance de Minkowski (généralisation de la distance) :

$$D(\mathbf{x}, \mathbf{y}) \stackrel{def}{=} \left( \sum_{i=1}^m |x_i - y_i|^r \right)^{\frac{1}{r}}$$

Équation 3

On notera que la distance euclidienne et la distance de Manhattan sont des cas particuliers de la distance de Minkowski.

Voici quelques fonctions plus complexes à présent :

Distance de Mahalanobis [35] :

$$D(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \det(\mathbf{V})^{-\frac{1}{m}} (\mathbf{x} - \mathbf{y})^T \mathbf{V} (\mathbf{x} - \mathbf{y})$$

Équation 4

où  $\mathbf{V}$  est la matrice de covariance de l'ensemble  $\{\mathbf{a}_1, \dots, \mathbf{a}_m\}$  avec  $\mathbf{a}_j$  le vecteur de valeurs du  $j^{\text{ème}}$  paramètre mesuré parmi les  $n$  ensembles de mesures de référence. Cette distance est requise lorsqu'une indépendance au facteur d'échelle est désirée.

Distance de corrélation :

$$D(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \frac{\sum_{i=1}^m (x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_{i=1}^m (x_i - \bar{x}_i)^2 \sum_{i=1}^m (y_i - \bar{y}_i)^2}}$$

Équation 5

avec  $\bar{x}_i$  et  $\bar{y}_i$  la valeur moyenne du  $i^{\text{ème}}$  attribut dans l'ensemble de mesures de référence.

La *métrique de valeur de différence* est un cas particulier, destinée à gérer notamment les attributs nominaux [31] :

$$D(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \sum_{c=1}^C \left| \frac{N_{a,x,c}}{N_{a,x}} - \frac{N_{a,y,c}}{N_{a,y}} \right|^q = \sum_{c=1}^C |P_{a,x,c} - P_{a,y,c}|^q$$

Équation 6

où :

- $N_{a,x}$  est le nombre d'instances dans l'ensemble de mesures de référence  $T$  ayant la valeur  $x$  pour l'attribut donné  $a$  ;
- $N_{a,x,c}$  est le nombre d'instances dans  $T$  ayant la valeur  $x$  pour l'attribut donné  $a$  et appartenant en sortie à la classe  $c$  ;
- $C$  est le nombre de classes présentes dans l'ensemble des mesures de référence (relatif au problème posé) ;
- $q$  est une constante (usuellement 1 ou 2) ;
- $P_{a,x,c}$  est la probabilité conditionnelle que la classe de sortie soit  $c$  lorsque l'attribut  $a$  à la valeur  $x$  (encore noté  $P(c | x_a)$ ).

Nous noterons que :

$$N_{a,x} \stackrel{\text{def}}{=} \sum_{c=1}^C N_{a,x,c}$$

## Équation 7

Avec une telle distance, deux valeurs sont considérées comme proches si elles ont une classification similaire en sortie. Par exemple, si l'attribut *couleur* possède trois valeurs {rouge, vert, bleu} et que l'application doit reconnaître une pomme, *rouge* et *vert* seront considérées comme étant plus proches que *rouge* et *bleu* étant donné que *rouge* et *vert* ont une corrélation similaire avec la classe de sortie *pomme*.

La métrique de valeur de différence exhibe un problème concernant les attributs ayant des valeurs qui n'ont jamais été présentes dans  $T$ . Dans ce cas, si  $a$  n'a jamais eu la valeur  $x$ , il s'ensuit que  $N_{a,x,c} = N_{a,x} = 0$  quel que soit  $c$ . Dans ce cas  $P_{a,x,c}$  n'est bien sûr plus défini ( $0/0$ ). Pour les valeurs nominales on suppose généralement simplement que  $P_{a,x,c} = 0$ . Le problème est plus complexe pour les attributs continus puisqu'ils peuvent prendre des valeurs très diverses, qui ne seront certainement pas toutes rencontrées dans  $T$ . Plusieurs solutions existent [32], consistant essentiellement à discrétiser les valeurs continues mais au prix d'une perte d'information. Par exemple, on peut considérer comme égales des valeurs continues comprises entre deux bornes mais une perte de précision survient puisqu'on considère comme égales les deux extrêmes de cet ensemble borné [33]. Contourner ces défauts demande encore d'autres techniques qui complexifient l'évaluation de la distance : ce sont les *métriques de valeur de différence interpolée, hétérogène, fenêtrée et discrétisée* [21].

Une distance permet donc de mesurer l'« éloignement » d'un objet à un objet de référence (par exemple dans le cas de la reconnaissance par motif). Elle possède souvent peu de signification intuitive, mais elle permet néanmoins de classer des objets si elle est considérée de manière relative. C'est donc un élément prépondérant quelle que soit l'approche envisagée. Nous allons à présent découvrir des méthodes plus sophistiquées, mais qui reposent malgré tout sur l'utilisation de ces fonctions.

### 3.3. La logique floue.

Un grand nombre de problèmes est caractérisé par le fait que les informations sont vagues, incomplètes ou imprécises. Un robot plongé dans un environnement hostile pourra par exemple renvoyer de mauvaises informations visuelles qu'il faut néanmoins traiter. Bien que surtout utilisée dans les systèmes de contrôle comme le freinage automobile avec ABS, le pilote automatique pour les avions ou encore les appareils photographiques, la logique floue peut aussi s'appliquer au problème de la reconnaissance de formes. En proposant des degrés de ressemblance et non des décisions binaires, cette approche nous a semblé intéressante à développer dans ce chapitre.

La logique floue fut introduite par Lofti A. Zadeh en 1965 [10] [19], et se base sur le concept d'*ensembles flous*. Un ensemble flou permet à un élément de lui appartenir à un degré particulier. Les conditions suivantes déterminent l'utilisation de la logique floue [17] :



- au moins un paramètre est continu ;
- un modèle mathématique du processus observé n'est pas connu ou ne peut pas être implanté en pratique (contraintes de mémoire, de puissance de calcul...etc.)
- les capteurs mis en jeu n'ont pas une précision très grande, ce qui rendrait l'utilisation d'un modèle mathématique illusoire.

Le processus de logique floue se déroule ainsi :

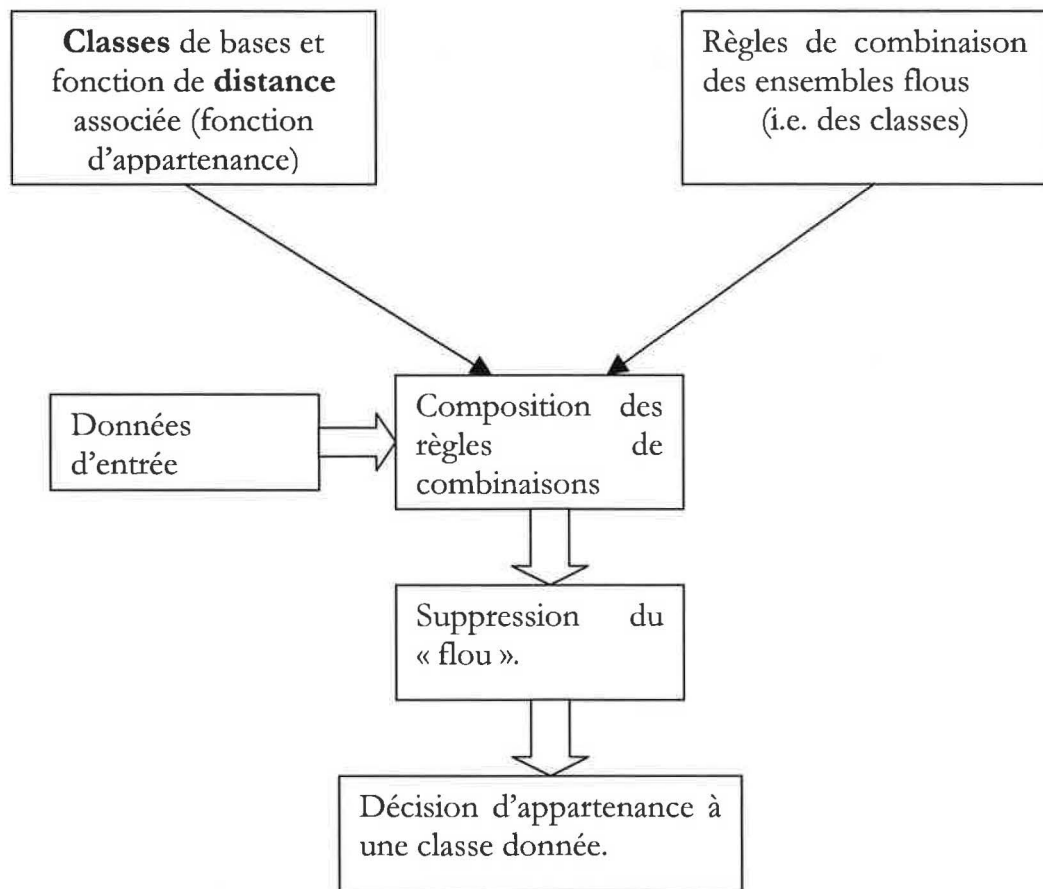


Figure 7

- **Les classes de bases** : ce sont les différents objets que le système peut reconnaître par exemple, ou plus précisément les valeurs attendues sur des attributs particuliers pour chaque objet distinct. Chaque attribut possède une **fonction de distance** associée. Ces classes sont considérées comme des **ensembles flous**.
- **Les règles de combinaison** : en fonction des résultats des fonctions de distance de chaque attribut, elles détermineront les degrés d'appartenance de l'objet observé (à travers les données d'entrée) à chaque ensemble flou.
- **La composition des règles de combinaisons** : le résultat est un ensemble de degrés d'appartenance à chaque classe présente.

- **La suppression du flou** : cette phase consiste à décider en fonction de la liste des degrés d'appartenance du rattachement de l'objet observé à une classe unique. Différentes techniques existent comme la maximisation de critère, la moyenne des maxima ou le centre de gravité des degrés d'appartenance.

La phase de suppression du flou peut ne pas être présente, et requiert alors la présence d'un expert pour prendre la décision. Cette phase est en effet très délicate et il n'existe pas de procédure systématique pour la résoudre.

Chaque étape soulève donc de multiples problèmes qui peuvent s'avérer complexes. Enfin, l'efficacité d'une telle méthode reste difficile à évaluer à large échelle.

La logique floue est conseillée pour :

- des problèmes complexes qui ne comportent pas de modèles mathématiques simples ;
- des processus fortement non-linéaires ;
- des processus où une expertise doit être réalisée, notamment avec des attributs définis de manière linguistique : plutôt grand, assez grand, moyen...etc.

Elle ne l'est pas si :

- les méthodes conventionnelles de mesure donnent de bons résultats ;
- un modèle mathématique existe ;
- le problème ne peut être résolu par d'autres méthodes.

### **3.4. Les réseaux neuronaux (reconnaissance statistique).**

Les réseaux neuronaux artificiels (RNA) [22] essaient d'apporter une solution nouvelle en s'inspirant des réseaux de neurones qui existent dans la nature. Pourquoi ? La réponse réside dans les points suivants :

- gérer des données bruitées ou extraites de l'environnement est généralement complexe ;
- le parallélisme massif est souvent complexe à mettre en œuvre ;
- adapter un programme en fonction du contexte est une tâche lourde puisqu'elle nécessite le plus souvent une réécriture.

#### **3.4.1. Qu'est-ce que c'est ?**

Bien qu'il n'y ait pas de définition universelle d'un RNA, des éléments communs sont identifiables. Il s'agit de multiples unités de calcul (ou *neurones*) reliées en réseau, chaque unité pouvant posséder une mémoire locale. Dans le réseau circulent des données

numériques, entrées ou sorties des différents « neurones ». Certains RNAs s'inspirent de modèles biologiques réels – d'animaux en général – la motivation historique étant de produire des systèmes artificiels capables de calculs sophistiqués, et peut-être « intelligents ».

Les RNAs nécessitent une *période d'apprentissage* qui va permettre leur calibration pour une tâche donnée. En d'autres termes, le RNA apprend à partir d'exemples, tout comme le ferait un jeune enfant, toute proportion gardée. Après la période d'apprentissage, le réseau est doté de capacités de généralisation qui s'appliquent à toute donnée relative au problème concerné.

Voici deux définitions issues de la littérature :

« Un Réseau de Neurones est un système massivement parallèle distribué ayant une propension naturelle à emmagasiner une connaissance issue de l'expérience. Il ressemble au cerveau en deux points :

- la connaissance est acquise par le réseau lors d'une phase d'apprentissage
- la pondération des connexions inter-neurales (*poids synaptiques*) stocke la connaissance. » [18]

« Une réseau de neurones est un circuit composé d'un grand nombre d'unités simples organisé de manière neurale. Chaque unité opère uniquement localement, de manière asynchrone. » [34]

### 3.4.2. Un exemple simple.

Considérons le réseau simple suivant :

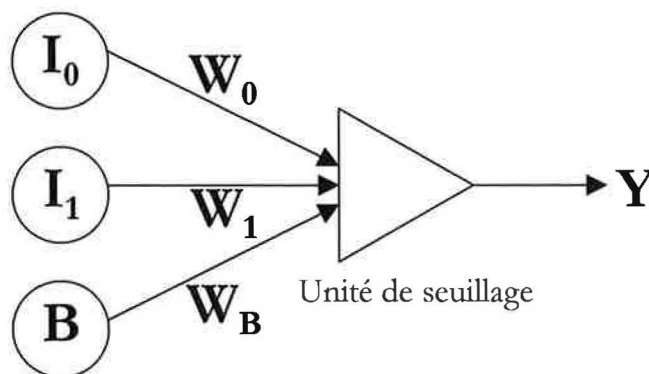


Figure 8

$I_0$ ,  $I_1$  et  $B$  sont les entrées.  $W_0$ ,  $W_1$  et  $W_B$  sont les pondérations des connexions. L'unité de seuillage est employée ici du fait de la nature continue des valeurs, et se définit comme suit :

- Si  $\mathbf{W}_0 \cdot \mathbf{I}_0 + \mathbf{W}_1 \cdot \mathbf{I}_1 + \mathbf{W}_B > 0$  Alors  $\mathbf{Y} = 1$
- Sinon  $\mathbf{Y} = 0$

Le but recherché est de faire apprendre au réseau neuronal la fonction logique OU (OR), c'est-à-dire renvoyer 1 si l'une des deux entrées  $\mathbf{I}_0$  ou  $\mathbf{I}_1$  est à 1.

L'apprentissage se fait de la manière suivante : changer les poids  $\mathbf{W}_0$ ,  $\mathbf{W}_1$  et  $\mathbf{W}_B$  d'un montant proportionnel à la différence entre la sortie voulue et la sortie réelle du système. L'équation est la suivante :

$$\Delta \mathbf{W}_i = \zeta \cdot (\mathbf{D} - \mathbf{Y}) \cdot \mathbf{I}_i$$

où  $\zeta$  est le *taux d'apprentissage*,  $\mathbf{D}$  est la sortie désirée (qui est fonction de ce qu'on veut apprendre au réseau neuronal) et  $\mathbf{Y}$  est la sortie réelle du réseau neuronal. On appelle cette loi *règle d'apprentissage du Perceptron* qui date des années 60.

On impose donc comme règle (table logique du OU) :

$\mathbf{I}_0$	$\mathbf{I}_1$	$\mathbf{D}$
0	0	0
0	1	1
1	0	1
1	1	1

Tableau 1

On expose ensuite le réseau à tous les états possibles de  $\mathbf{I}_0$  et  $\mathbf{I}_1$ . Les pondérations sont ajustées à chaque essai, jusqu'à obtention de l'égalité :

$$\mathbf{D} - \mathbf{Y} = 0$$

Équation 8

De manière générale, le processus d'apprentissage est le suivant :

- marche avant : les sorties sont calculées ainsi que l'erreur commise par rapport à la sortie désirée ;
- marche arrière : l'erreur est employée pour modifier les pondérations des connexions. Une première étape altère les pondérations en sortie, une seconde étape rétropropage l'erreur à travers tout le système.

Ce processus est itéré jusqu'à ce que l'erreur soit nulle ou inférieure à un certain seuil.

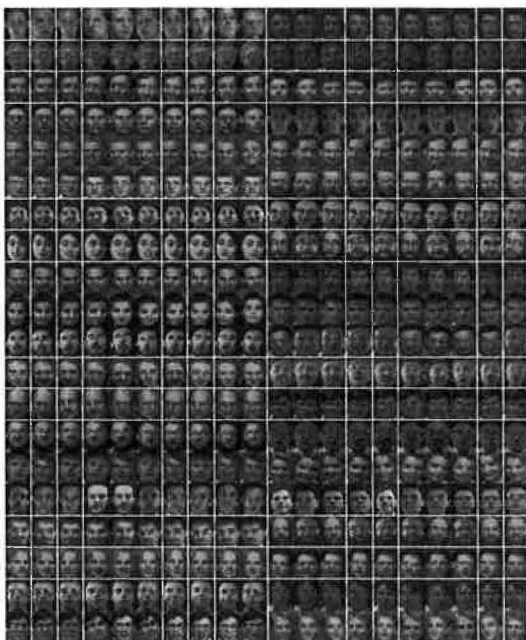
### 3.4.3. Perfectionnements et déboires des réseaux de neurones.

Des méthodes de co-évolution existent, dans lesquelles sont placés en compétitions plusieurs réseaux de neurones, le meilleur étant retenu. Les poids synaptiques des RNAs sont encodés dans des génotypes, et une heuristique d'optimisation par algorithme génétique est opérée. C'est Samuel qui introduisit ce concept dans les années 50-60 pour son programme de jeux de dames.

Les RNAs furent mis en péril lorsque Papert et Minski prouvèrent que le *Perceptron* ne pouvait pas déterminer la connexité d'une région dans une image binaire. C'est la découverte de la méthode de rétropropagation qui remit au goût du jour les réseaux de neurones.

#### 3.4.4. Quand utiliser les RNAs ? Avantages et inconvénients.

Le principal avantage du RNA est qu'il s'emploie lorsqu'aucune solution algorithmique n'est connue, ou qu'il est difficile d'extraire des critères pertinents à mesurer pour la reconnaissance de formes [23] :



Voici une base de donnée typique de visages pour un réseau de neurones. Chaque personne possède plusieurs vues, avec de légères différences. On apprend ainsi au réseau à reconnaître des personnes de manière automatique en lui soumettant de vastes bibliothèques d'images.

Des zones comme les yeux ou la bouche sont spécifiées, et vont constituer les éléments pertinents à analyser. Ces éléments sont envoyés au RNA qui est doté de fonctions de mesure sur ces éléments. Celui-ci va combiner ces fonctions au mieux à partir de la base d'apprentissage jusqu'à obtenir un taux de succès maximal.

Avantages :

- le RNA peut être facilement parallélisé ;
- il régule lui-même les fonctions de discrimination des éléments pertinents à observer.

Il est important de remarquer que les RNAs ne sont pas plus puissants que toute autre méthode : ils ne peuvent résoudre des problèmes prouvés insolubles. Ils permettent cependant de traiter des cas qui seraient très complexes à mettre en œuvre autrement. En particulier, ils sont capables de créer un « modèle » à partir des données d'apprentissage sans que celui-ci soit explicitement formulé.

Inconvénients :

- réguler les pondérations lors de la phase d'apprentissage est coûteux en calculs. Les algorithmes de rétropropagation peuvent être lents à converger ;
- la taille des données d'apprentissage est difficile à estimer : si elle est trop faible, le système semblera opérer avec succès alors qu'il se sera simplement calibré sur ce jeu particulier de données ;
- on ne peut pas savoir si la solution idéale est contenue dans l'ensemble des données d'apprentissage du RNA, ce qui peut empêcher toute obtention de bons résultats ;
- le nombre de neurones à implanter est à déterminer ;
- on ne peut pas prouver qu'un RNA donnant satisfaction sur un jeu de données fournira toujours des résultats corrects.

#### **4. CONCLUSION.**

Il est indubitable que les algorithmes de reconnaissance de forme se basent sur des études concernant notre propre fonctionnement. Des modèles comme les réseaux artificiels de neurones ainsi que la terminologie employée sont autant de preuves de l'origine de ces travaux. Il reste que les processus biologiques mis en jeu restent mal connus, ce qui induit par là même un succès très limité des méthodes algorithmiques. Inversement, les tenants de la « psychologie computationnelle » proposent des modèles informatiques de la pensée humaine et les programment pour tester leur validité et détecter leurs limitations.

Un des aspects fondamentaux est notre capacité à reconnaître des objets toujours de la même manière malgré la variabilité d'angle de vue par exemple : si nous regardons différentes photographies d'un même objet pris sous des positions différentes, nous sommes néanmoins capables de le reconnaître. N'oublions pas que l'image que nous obtenons sur la rétine n'est qu'une projection en deux dimensions d'un objet tridimensionnel, ce qui induit forcément une perte d'information voire des ambiguïtés. Il semble évident que notre processus de compréhension des formes est le résultat d'une subtile combinaison d'au moins toutes les méthodes vues dans ce chapitre, puisque chaque approche apporte une partie de la solution. Pourtant il manque certainement d'autres pierres à l'édifice : comment sont stockés les objets dans le cerveau ? Comment pouvons-nous reconnaître le sens d'un discours même dans un poste de radio de mauvaise qualité ? Comment filtrons-nous tous les stimuli qui arrivent instantanément et constamment dans tous nos canaux sensoriels ? Toutes ces questions sont le théâtre d'une multitude de travaux de recherche en biologie, neurologie, psychologie cognitive et... informatique, preuve s'il en est de l'énorme potentialité d'un tel secteur mais aussi de l'énorme frustration qu'il a engendré ces vingt dernières années quant à son appréhension.

## **5. HISTORIQUE DE L'INTELLIGENCE ARTIFICIELLE DE 1637 À 1992 : MYTHES ET RÉALITÉS...**

- 1637 : Descartes « Je pense, donc je suis. »
- 1726 : Les Voyages de Gulliver de Jonathan Swift : une machine qui écrit des livres
- 1811 : Mouvement de Ludd contre l'automatisation du travail (Luddisme)
- 1847 : Boole développe la logique symbolique puis plus tard la logique binaire
- 1859 : Publication de « L'Origine des Espèces » par Darwin
- 1879 : Frege invente le calcul de prédicats
  
- 1901 : « L'interprétation des Rêves » de Sigmund Freud
- 1917 : Karel Capek cite le terme « robot » (en Tchèque, robot signifie travailleur)
- 1930 : Shannon démonte la logique Booléenne avec des circuits à commutation
- 1930 : Vannevar Bush construit un analyseur différentiel analogique au MIT
- 1931 : Théorème de l'incomplétude de Gödel
- 1937 : Turing publie « *On Computable Numbers* » (modèle de Machine de Turing)
- 1939 : Dickinson dépose un brevet sur un composant électronique de stockage
- 1940 : Atanasoff et Berry construisent le 1<sup>er</sup> ordinateur électronique, baptisé ABC
- 1940 : Robinson est le premier ordinateur opérationnel en Angleterre, utilisé pour décoder le cryptage Nazi
- 1941 : Zuse construit le Z3 en Allemagne, c'est le 1<sup>er</sup> ordinateur programmable
- 1943 : « Colossus » utilise des tubes électroniques pour aider les anglais à décrypter les codes allemands
- 1943 : McCulloch et Pitts proposent une architecture par « réseau de neurones » pour l'intelligence
- 1944 : Aiken fabrique le 'Mark I', 1<sup>er</sup> ordinateur programmable américain
- 1945 : Grace Murray Hopper découvre le premier « bug » le 9 septembre 1945 à 15:45
- 1945 : Vannevar Bush publie « *As we may think* » dans l'*Atlantic Monthly*
- 1946 : Eckert et Mauchley construisent l'ENIAC, le 1<sup>er</sup> ordinateur programmable numérique
- 1948 : Norbert Wiener écrit « *Cybernetics* »
- 1949 : Orwell publie « 1984 », où les ordinateurs sont employés pour contrôler les personnes
- 1949 : Shannon introduit la Théorie de L'Information
- 1950 : Isaac Asimov, « I, Robot »
- 1950 : Shannon propose un programme jouant aux échecs
- 1950 : Test de Turing (« *Computing Machinery and Intelligence* »)
- 1950 : Eckert & Mauchley lancent l'UNIVAC, 1<sup>er</sup> ordinateur commercial
- 1952 : Mort d'Alan Turing
- 1952 : CBS emploie UNIVAC pour prédire l'élection d'Eisenhower-Stevenson
- 1952 : 1<sup>er</sup> ordinateur utilisé par le *Department of Defense* (IBM 701)
- 1952 : Grace Murray Hopper décrit un compilateur
- 1953 : Watson et Crick découvrent la structure chimique de l'ADN
- 1954 : Isaac Asimov, « *The Caves of Steel* »
- 1955 : Newell, Shaw et Simon développent l'IPL-11, le premier langage pour l'IA



- 1955 : IBM introduit le premier calculateur à transistors
- 1956 : Newell, Shaw et Simon créent « *The Logic Theorist* », résolvant des problèmes mathématiques
- 1956 : le terme « *AI* » est employé à la *Dartmouth computer conference*, première rencontre entre McCarthy, Minsky, Newell et Simon
- 1956 : CIA fonde le GAT, projet de machine traductrice
- 1956 : Ulam développe « *MANIAC I* », le premier programme de jeu d'échec qui bat l'homme
- 1957 : Chomsky écrit « *Syntactic Structures* »
- 1957 : Newell, Shaw et Simon créent le GPS (*General Problem Solver*)
- 1958 : McCarthy introduit le « *LISP* » au MIT
- 1958 : fondation du DARPA
- 1958 : Jack St. Clair Kilby invente le circuit intégré
- 1959 : Minsky et McCarthy fondent le laboratoire d'IA au MIT
- 1959 : Rosenblatt introduit le « *Perceptron* »
- 1959 : Le jeu de dames de Samuel bat le meilleur joueur humain
- 1959 : DEC annonce le PDP-1 à \$159 000
- 1960 : L'ordinateur du DoD prend la lune pour un missile
- 1960 : LINC est le 1<sup>er</sup> mini-ordinateur, il est équipé d'un tube cathodique (*Lincoln Labs*)
- 1960 : Lecteur de bande
- 1960 : Bar-Hillel publie un article sur les difficultés de la traduction automatique
- 1961 : DEC vend le PDP-1 à \$120 000
- 1962 : McCarthy intègre Stanford, où il fonde le *Stanford AI Lab* en 1963
- 1962 : Purdue fonde le premier département d'informatique proposant des thèses de doctorat
- 1962 : Premiers robots industriels commercialisés
- 1963 : l'ARPA investit 2 millions de dollars au *MIT AI Lab*
- 1963 : M. Ross Quillian introduit les réseaux sémantiques pour la représentation des connaissances
- 1963\* Susumo Kuno teste son analyseur syntaxique sur « *Time flies like an arrow* », mettant en évidence l'ambiguïté du langage
- 1963 : Minsky, « *Steps Towards Artificial Intelligence* »
- 1964 : le DEC PDP-8 est le 1<sup>er</sup> ordinateur produit industriellement
- 1964 : Kemeny et Kurtz introduisent le « *BASIC* »
- 1964 : Bobrow programme « *STUDENT* » qui résout des problèmes d'algèbre de lycée
- 1965 : Buchanan, Feigenbaum et Lederberg fondent le projet de système expert « *DENDRAL* »
- 1965 : Simon prédit « *by 1985 machines will be capable of doing any work a man can do* »
- 1965 : Dreyfus dénie les possibilités d'IA
- 1966 : Donald Michie fonde l'*Edinburgh AI lab*
- 1967 : le programme « *MacHack* » de Greenblatt bat Hubert Deyfus aux échecs
- 1967 : IBM sépare le *hardware* du *software*
- 1968 : « *HAL* » est la vedette du film de Clarke et Kubrick, « *2001* »
- 1968 : Première thèse en informatique (Wexelblat à l'Université de Pennsylvanie)
- 1968 : Minsky, « *Semantic Information Processing* »

- 1968 : Chomsky et Halle, « *The Sound Pattern of English* »
- 1969 : la thèse d'Alan Kay décrit théoriquement le fonctionnement de l'ordinateur personnel (PC)
- 1969 : Minsky et Papert, « *Perceptrons* » (limites de réseau de neurones simple couche)
- 1969 : UNIX (Thomson et Ritchie à AT&T)
- 1969 : Hearn et Griss définissent le *Standard Lisp* pour traduire le système de résolution formelle algébrique « *REDUCE* »
- 1970 : PROLOG (Colmerauer)
- 1970 : introduction de la disquette
- 1970 : Pople et Myers démarrent « *INTERNIST* », un programme d'assistance au diagnostic médical
- 1971 : 1<sup>er</sup> microprocesseur aux Etats-Unis (Intel 8008)
- 1971 : 1<sup>er</sup> calculateur de poche (*Pocketronic*)
- 1971 : langage Pascal
- 1972 : Dreyfus publie « *What Computers Can't Do* »
- 1973 : le rapport de Lighthill annule le financement de l'IA en Grande Bretagne
- 1973 : Distribution d'UNIX aux universités
- 1974 : 1<sup>er</sup> robot piloté par ordinateur
- 1974 : Minsky, « *A Framework for Representing Knowledge* »
- 1974 : établissement du réseau SUMEX-AIM (application de l'IA à la médecine)
- 1975 : Cooper et Erlbaum fondent « *Nestor* » pour développer la technologie par réseau de neurones
- 1975 : 1<sup>er</sup> BASIC pour micro-ordinateur par Gates et Allen
- 1975 : 1<sup>er</sup> ordinateur personnel Altair 8800 (256 octets de mémoire)
- 1975 : le DARPA lance le programme de reconnaissance d'images
- 1975 : Larry Harris fonde *Artificial Intelligence Corp*
- 1976 : « *Adventure* » (Crowther et Woods), premier jeu d'aventure
- 1976 : Greenblatt crée la première machine LISP, « *CONS* »
- 1976 : Kurzweil introduit la machine lectrice
- 1976 : super-ordinateur Cray-1, 138 mégaflops
- 1976 : Dynabook paper (Kay and Goldberg)
- 1977 : 3CPO et R2D2 apparaissent dans le film « *Star Wars* » de George Lucas
- 1979 : introduction de « *Pac Man* »
- 1979 : Raj Reddy fonde le « *Robotics Institute* » à l'université Carnegie Mellon
- 1979 : le programme MYCIN aussi bon qu'un expert médical (*Journal of American Medical Assoc.*)
- 1980 : les systèmes experts atteignent un millier de règles
- 1980 : première conférence en IA à Stanford
- 1980 : première conférence bi-annuelle *ACM Lisp and Functional Programming Conference*
- 1980 : Xerox, DEC et Intel annoncent *Ethernet*
- 1981 : IBM introduit le *Personal Computer* (PC)
- 1981 : MITI prédit l'ordinateur intelligent d'ici 1990
- 1982 : Publication du « rapport Alvey » en Grande Bretagne sur les technologies avancées de l'information, qui conduira à l'emploi de l'IA dans le domaine de l'industrie
- 1982 : John Hopfield ressuscite les réseaux de neurones

- 1983 : Asimov, « *Robots of Dawn* »
- 1983 : 6,000,000 d'ordinateurs vendus
- 1983 : démantèlement d'AT&T
- 1983 : Sony annonce la technologie « *compact disc* »
- 1984-86 : les industriels investissent 50 millions de dollars dans les « startups » AI
- 1984 : le programme « *RACTER* » de Chamberlain « écrit » des livres
- 1984 : la Communauté Européenne fonde le programme ESPRIT
- 1984 : General Motors investit 4 millions de dollars dans la société *Teknowledge*
- 1984 : Texas Instruments remporte le contrat au MIT pour la machine LISP
- 1984 : « *Wabot-2* » lit des partitions et joue de l'orgue
- 1984 : Apple introduit le *Macintosh*
- 1984 : IBM introduit le composant RAM à 1 mégabit
- 1984 : introduction des disques optiques
- 1985 : le système expert « *Soup* » de GM et Campbell n'emploie pas le LISP
- 1985 : un robot chez Kawasaki tue un mécanicien Japonais suite à un dysfonctionnement
- 1985 : Minsky publie « *The Society of Mind* »
- 1985 : *Teknowledge* abandonne le LISP au profit du PROLOG et du C
- 1985 : Xerox emporte un contrat de 20 millions de dollars pour des machines LISP, qui est ensuite annulé
- 1985 : C++
- 1986 : revenu de l'IA estimé à 1 milliard de dollars
- 1986 : le robot joueur de ping-pong d'Anderson bat l'homme
- 1986 : la « *HiTech chess machine* » de CMU joue en niveau senior
- 1986 : la police de Dallas emploie un robot pour pénétrer dans un appartement
- 1986 : IBM entre dans l'IA à l'AAAI
- 1986 : des startups liées aux réseaux de neurones apparaissent
- 1986 : la reconnaissance automatique de caractères (OCR) est une industrie évaluée à 100 millions de dollars de revenus
- 1987 : on dénombre 1,900 systèmes experts dans le monde
- 1987 : l'AI rapporte 1,4 milliards de dollars, sans compter la robotique
- 1987 : la vision robotique rapporte 300 millions de dollars
- 1987 : système AFIS Japonais (identification d'empreintes digitales)
- 1987 : « Hiver de l'IA » ; saturation du marché des machines LISP
- 1988 : le processeur i386 (intel) met les machines LISP à la portée des PC
- 1988 : les systèmes experts rapportent 400 millions de dollars
- 1988 : Minsky et Papert publie une version remaniée de « *Perceptrons* »
- 1988 : TI annonce « *microExplorer* », un Macintosh avec processeur LISP
- 1989 : arrêt de fonctionnement de plus d'un millier de systèmes informatiques hospitaliers américains (dépassement de date codée sur 16 bits depuis le 1/1/1900)
- 1990 : doublement du budget du programme ESPRIT
- 1992 : Apple Computer introduit « *Dylan* », un langage basé sur le LISP, comme une vision du futur de la programmation

## 6. BIBLIOGRAPHIE.

- [1] P. Ferraro, C. Godin, « *Une Méthode de Comparaison Structurale d'Arborescences non Ordonnées* », Document de Travail du Programme Modélisation des Plantes N°1-98, CIRAD, Avril 1998
- [2] C. Berge, « *Graphs* », North Holland, 3<sup>rd</sup> Edition, 1991
- [3] M. Gondran, M. Minoux, « *Graphes et Algorithmes* », Eyrolles, 1990
- [4] S.-Y. Lu, « *A tree-to-tree Distance and its Applications to Cluster Analysis* », IEEE Transactions on PAMI, 1(2), pp. 219-224, 1979
- [5] L. Miclet, « *Méthodes Structurelles pour la Reconnaissance de Formes* », Eyrolles, Paris, 1984
- [6] K. Zhang, « *A Constrained Edit Distance between Unordered Labeled Trees* », Algorithmica, 15, pp. 205-222
- [7] A. G. Mamistvalov, « *n-Dimensional Moment Invariants and Conceptual Mathematical Theory of Recognition n-Dimensional Solids* », IEEE Transactions on PAMI, 20(8), pp. 819-831
- [8] D. L. Borges, R. B. Fisher, « *Class-Based Recognition of 3D Objects Represented by Volumetric Primitives* », British Machine Vision Conference, 1996
- [9] D. L. Borges, « *Recognizing Three-Dimensional Objects Using Parameterized Volumetric Models* », PhD. Thesis, Dept. Artificial Intelligence, University of Edinburgh, 1995
- [10] L. A. Zadeh, « *Fuzzy Sets as a Basis for a Theory of Possibility* », Fuzzy Sets and Systems, 1(1), pp. 3-28, 1978
- [11] I. Biederman, « *Recognition-by-Components : A Theory of Human Image Understanding* », Psychological Review, 94, pp. 115-147, 1987
- [12] M. Näf, O. Kübler, R. Kikinis, M. E. Shenton and G. Székely, « *Characterization and Recognition of 3D Organ Shape in Medical Image Analysis Using Skeletonization* », IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, Hotel Nikko, San Francisco, 1996
- [13] S. Carlsson, « *View Variation and Linear Invariants in 2D and 3D* », Computational Vision and Active Perception Laboratory, Stockholm, Sweden, TRITA – NA – P 9522, CVAP 184
- [14] S. Sclaroff, « *Deformable Prototypes for Encoding Shape Categories in Image Databases* », Pattern Recognition, 30(4), Special Issue on Image Database, April 1997
- [15] D. Forsyth, J. L. Mundy, A. Zisserman, C. Coelho, A. Heller and C. Rothwell, « *Invariant Descriptors for 3D Object Recognition and Pose* », IEEE Transactions on PAMI, 13(10), pp. 971-991, October 1991
- [16] R. Alferez, Y.-F. Wang, « *Geometric and Illumination Invariants for Object Recognition* », IEEE Transactions on PAMI, 21(6), pp. 505-535, June 1999
- [17] E. Cox, « *The Fuzzy Systems Handbook* », Academic Press, 1994
- [18] S. Haykin, « *Neural Networks* », MacMillan, 1994
- [19] L. A. Zadeh, « *Fuzzy Sets* », Information and Control, 8, pp. 338-353, 1965
- [20] S. Sclaroff and A. Pentland, « *Modal Matching for Correspondence and Recognition* », IEEE Transactions on PAMI, June 1995
- [21] D. R. Wilson, T. R. Martinez, « *Improved Heterogeneous Distance Functions* », Journal of Artificial Intelligence Research, 6, pp. 1-34, January 1997
- [22] R. Hecht-Nielsen, « *Neurocomputing* », Reading, MA : Addison-Wesley
- [23] B. D. Ripley, « *Pattern Recognition and Neural Networks* », Cambridge University Press

- [24] P. Schyns, R. L. Goldstone and J.-P. Thibaut, « *The Development of Features in Object Concepts* », Behavioral and Brain Sciences, 21(1), pp. 1-54, 1998
- [25] M. D. Adler, « *An Introduction to Pattern Recognition : Statistical, Neural Net and Syntactic Methods of Getting Robots to See and Hear* », <http://ciips.ee.uwa.edu.au/~mike/PatRec/PatRec.html>, 1997
- [26] T. M. Mitchell, « *The Need for Biases in Learning Generalizations* », Readings in Machine Learning, J. W. Shavlik and T. G. Dietterich Editors, 1980
- [27] C. Schaffer, « *A Conservation Law for Generalization Performance* », Proceedings of the Eleventh International Conference on Machine Learning (ML'94), Morgan Kaufmann, 1994
- [28] P. Turney, « *Exploiting Context when Learning to Classify* », Proceedings of the European Conference on Machine Learning, pp. 402-407, Vienna, Austria, Springer-Verlag, 1993
- [29] D. H. Wolpert, « *On Overfitting Avoidance as Bias* », Technical Report SFI TR 92-03-5001, Santa Fe, The Santa Fe Institute, 1993
- [30] D. Wettscherek, D. W. Aha and T. Mohri, « *A Review and Comparative Evaluation of Feature Weighting Methods for Lazy Learning Algorithms* », Technical Report AIC-95-012, Washington, DC, Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence, 1995
- [31] C. Stanfill, D. Waltz, « *Toward Memory-Based Reasoning* », Communications of the ACM, 29, pp. 1213-1228, December 1986
- [32] D. Ventura, « *On Discretization as a Preprocessing Step for Supervised Learning Models* », Master's Thesis, Department of Computer Science, Brigham Young University
- [33] D. Ventura, T. R. Martinez, « *An Empirical Comparison of Discretization Methods* », Proceedings of the Tenth International Symposium on Computer and Information Sciences, pp. 443-450, 1995
- [34] A. Nigri, « *Neural Networks for Pattern Recognition* », p. 11, Cambridge, MA : The MIT Press, 1993
- [35] P.C. Mahalanobis. On the Generalized Distance in Statistics. *National Institute of Science in India*, 12:49-55, 1936.
- [36] P. Devijver, J. Kittler, « *Pattern Recognition. A statistical Approach.* », Prentice Hall 82.
- [37] G. Gaillat, « *Méthodes statistiques de Reconnaissance des Formes.* », ENSTA 83.
- [38] L. Miclet, « *Méthodes structurelles pour la Reconnaissance des Formes.* », Eyrolles 84.

## 5. Le E-Squelette.

*« La caractéristique fondamentale de l'intelligence, qu'elle soit qualifiée de naturelle ou d'artificielle est l'interprétation. Une forme ou configuration étant perçue, que va-t-elle provoquer ? Sera-t-elle mise en mémoire ? Pour interpréter, il faut classer, séparer, comparer ; ceci dans une diversité presque incalculable. Bien entendu, on ne perçoit pas le réel, mais une partie de ce réel auquel on est sensible, directement ou par l'intermédiaire de machines concrètes ou abstraites. Ce qui est perçu il faut le séparer en classes, en familles, en agrégats, à partir desquels l'interprétation pourra se réaliser. »*

A. Kaufmann, 1975

## **1. INTRODUCTION.**

Nous présentons dans ce chapitre le modèle de représentation d'objets volumiques que nous avons baptisé *Squelette Ellipsoïdal* ou *E-Squelette* [1][2]. C'est essentiellement une chaîne de méthodes mathématiques et algorithmiques organisées de manière modulaire : chaque maillon peut être modifié pour satisfaire à des demandes particulières. Nous avons bien sûr fait des choix pour mettre en place le E-Squelette, et malgré les justifications apportées à chacun d'eux, ils ne sont en aucun cas rigides. Certains aspects induisent de nouvelles pistes de recherche tant sur le plan technique que conceptuel.

Les motivations principales du E-Squelette sont de proposer une représentation géométrique compacte, dynamiquement adaptable en qualité selon les besoins, tout en caractérisant le mieux possible un solide tridimensionnel. Regrouper ces critères est complexe car un bon modèle géométrique est peut-être un très mauvais identifiant et inversement. Certains choix sont certainement discutables et perfectibles, c'est pour cela que nous proposons une architecture ouverte.



## 2. MOTIVATIONS.

Elles s'inscrivent dans le thème de la *représentation des données* non seulement en tant que support de l'information, avec toutes les problématiques de visualisation que cela suppose, mais aussi de synthèse de l'information. Ce dernier aspect est une passerelle vers les techniques de reconnaissance automatique de formes, car il sous-entend une caractérisation satisfaisante de chaque entité. Si l'on veut rendre l'information plus compacte, il est en effet primordial d'en tirer « la substantifique moelle » pour reprendre la célèbre expression de François Rabelais. Si les techniques mises en œuvre sont adéquates, elles fourniront une carte d'identité de l'objet [9][14]. Il peut être de surcroît intéressant de réguler le filtrage de l'information, de manière à contrôler sa précision. A faible précision l'objet est mal décrit mais sa « forme générale » est perceptible, ce qui peut conduire à des possibilités d'appariement plus larges. A haute précision, l'objet peut-être étudié de manière complète mais il présente alors trop de caractères particuliers pour être apparié à d'autres, phénomène souvent lié à un trop grand morcellement de l'information.



Figure 1

Observons cet ensemble de cactus (*Joshua Trees*). L'image de gauche permet à l'être humain de reconnaître les arbustes entre eux car, les regardant d'assez loin, nous ne percevons pas les différences propres à chacun d'eux. Si nous nous en approchons (image de droite), les caractères particuliers émergent : les branches ne seront pas disposées identiquement, les tailles seront différentes, voire les couleurs...Etc. En contrôlant le *niveau sémantique*, le E-Squelette autorise des appariements normalement impossibles de manière classique.

Le but du E-Squelette est donc de proposer un modèle flexible de représentation géométrique. Les différents champs applicatifs visés sont tout naturellement :

- la reconnaissance automatique et hiérarchique d'objets ;
- la visualisation interactive par niveaux de détails ;
- la compression progressive des données.



### 3. FORMALISME ET OUTILS THÉORIQUES DU E-SQUELETTE.

Dans cette partie nous allons présenter non seulement une formalisation du E-squelette, mais aussi les outils mathématiques qui le constituent, en apportant les justifications nécessaires au travers de démonstrations.

#### 3.1. Définition formelle du E-Squelette.

On définit un *objet tridimensionnel*  $O$  par un ensemble de points  $P_r$  de l'espace représentés par les quadruplets :

$$(x_r, y_r, z_r, \omega_r)$$

où  $x_r$ ,  $y_r$  et  $z_r$  sont les coordonnées dans  $\mathbb{R}^3$  de  $P_r$  et  $\omega_r$  leur pondération dans  $\mathbb{R}$ .

Un *niveau sémantique*  $n$  du *squelette ellipsoïdal* (où *E-squelette*)  $\xi_n$  d'un objet tridimensionnel  $O$  est défini par l'ensemble :

$$\xi_n \stackrel{def}{=} \{C_1, \dots, C_{n+1}\}$$

Équation 1

où  $C_i (i \in \{1, \dots, n+1\})$  est une classe caractérisée par le vecteur  $(\alpha_1^i, \dots, \alpha_p^i)$  portant l'ensemble de paramètres  $\alpha_j^i \in \mathbb{R}^{k_j}$ , avec  $k_j$  entier naturel. Remarquons que  $\alpha_j^i$  peut aussi être une valeur discrète.

##### 3.1.1. Classes et sous-classes composant le E-Squelette.

Pour tout niveau sémantique  $n$ , les  $n+1$  classes  $C_i$  réalisent une *partition stricte* du nuage complet  $X$  de  $\mathbb{R}^3$  représentant l'objet  $O$ , c'est-à-dire vérifiant la propriété suivante :

$$\begin{aligned} \bigcup C_i &= X \\ C_i \cap C_j &= \emptyset \quad \forall i < j \leq n+1 \end{aligned}$$

Équation 2

Le niveau sémantique  $n$  définit le nombre de classes de la façon suivante : si  $n=0$  il y a une classe, si  $n=1$  il y a deux classes...etc. L'ensemble des partitions possibles est nommé *espace de recouvrement* de  $X$ .

Nous allons à présent introduire quelques notions de base de statistique afin de pouvoir aborder le processus d'obtention des sous-classes  $C_i$ .

La *variance* d'un ensemble  $X$  de  $m$  valeurs  $x_j$  pondérées par les réels  $\omega_j$  relativement à la moyenne pondérée  $\bar{x}$  est définie comme :

$$\text{var}(X) \stackrel{\text{def}}{=} \frac{\sum_{j=1}^m (x_j - \bar{x})^2 \omega_j}{\sum_{j=1}^m \omega_j}$$

Équation 3

où  $\omega_j$  est un réel positif ou nul pondérant la valeur  $x_j$ , et  $\bar{x}$  s'exprime ainsi :

$$\bar{x} \stackrel{\text{def}}{=} \frac{\sum_{j=1}^m \omega_j x_j}{\sum_{j=1}^m \omega_j}$$

Équation 4

Soit  $C_s = (C_1, \dots, C_{n+1})$  une partition stricte de  $X$  en  $n+1$  classes (niveau sémantique  $n$ ). La *variance intraclasse* de  $C_s$  est alors :

$$\text{var}_{\text{intra}}(C_s) \stackrel{\text{def}}{=} \text{var}(X) - \text{var}_{\text{inter}}(C_1, \dots, C_{n+1})$$

Équation 5

où  $\text{var}_{\text{inter}}(C_1, \dots, C_{n+1})$  est défini comme la *variance interclasse globale* de la partition stricte de  $X$  (niveau sémantique  $n$ ) :

$$\text{var}_{\text{inter}}(C_1, \dots, C_{n+1}) \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{n+1} \Omega_i (\bar{x}_i - \bar{x})^2}{\Omega}$$

Équation 6

avec  $\Omega_i = \sum_{x_j \in C_i} \omega_j$  (poids de la classe  $C_i$ ),  $\Omega = \sum_X \omega_j$  (poids de  $X$ ) et  $\bar{x}_i \stackrel{\text{def}}{=} \frac{\sum_{x_j \in C_i} \omega_j x_j}{\Omega_i}$

(moyenne de la classe  $C_i$ ).

Pour définir un partitionnement de  $X$ , il faut le munir d'une *structure de représentation* en lui associant un ensemble  $L$  et une application  $D$  de  $X \times L$  dans  $\mathbb{R}^+$ ,  $L$  regroupe les représentants de chaque classe, chaque classe ayant un seul représentant.

$L$  est l'espace de représentation, chacun de ses éléments est une représentation ; si  $(x, l) \in X \times L$ ,  $D(x, l)$  est appelée mesure de dissemblance de l'objet  $x$  à la représentation  $l$ .

Exemple :

Soit  $L = \mathbb{R}$ ,  $D(x, l)$  est la distance euclidienne dans  $\mathbb{R}$  de  $x$  à  $l$ .  $l$  est par exemple le centre de gravité dans  $\mathbb{R}$  d'une partie de  $X$ .

On appelle *fonction de représentation* la fonction  $g$  qui permet d'associer à toute partie de  $X$  une représentation unique appartenant à  $L$ . Si  $g(Q) = l$  on dit que  $l$  est une représentation de  $Q \subset X$ .

On appelle *fonction d'affectation* l'application  $f$  qui permet d'associer à tout élément de  $L$  une classe d'une des partitions de  $X$ .

Ces notions étant énoncées, nous allons à présent développer une méthode particulière de partitionnement nommée *nuées dynamiques* [13].

### 3.1.2. Construction des classes du E-Squelette : les nuées dynamiques.

Cet algorithme est aussi connu sous le nom de **K-Mean**, **K-moyenne**, **moyennes mobiles** ou encore **moyennes migratoires**.

Nous avons choisi cette technique de partitionnement pour les raisons suivantes :

- son cheminement est intuitif et prévisible du fait de l'utilisation de représentants de classe et de distances interprétables mécaniquement ou géométriquement. La méthode apporte ainsi une réponse compréhensible en termes de caractérisation ;
- sa convergence est très rapide et les effets de stagnation sont très réduits. Cette robustesse notoire autorise une utilisation générique des nuées dynamiques ;
- elle peut être facilement modifiée selon le domaine concerné et s'inscrit donc intégralement dans l'architecture ouverte qui est proposée.

La méthode des nuées dynamiques se déroule en plusieurs phases :

- choisir un espace  $S$  des recouvrements de  $X$  ;
- munir  $X$  d'une structure de représentation, c'est-à-dire choisir un espace  $L$  de représentation et une application  $D: X \times L \rightarrow \mathbb{R}^+$  ;
- définir un *critère d'optimisation*. C'est une application, notée  $\mathbf{W}$ , à valeurs positives, qui doit permettre en utilisant  $D$  de mesurer l'adéquation entre tout recouvrement et toute représentation de ce recouvrement ;

- définir un problème d'optimisation dont le but est la minimisation du critère  $\mathbf{W}$ . Ce problème s'énonce alors ainsi : « chercher simultanément le recouvrement  $S' \subset S$  et une représentation  $L$  de ce recouvrement de façon que  $S'$  et  $L$  aient la meilleure adéquation possible au sens du critère  $\mathbf{W}$  » ;
- construire un algorithme (dit des nuées dynamiques). Cet algorithme consiste à utiliser de manière itérative une fonction de représentation  $g$  et une fonction d'affectation  $f$ . Il est initialisé à l'aide d'un recouvrement  $S^{(0)}$  estimé ou tiré au hasard ou d'une partie de l'espace de représentation  $L^{(0)}$  également estimée ou tirée au hasard ;
- étudier les propriétés de convergence de la suite  $u_n = W(v_n)$  avec  $v_n = (S^{(n)}, L^{(n)})$  où  $S^{(n)} \in S$  et  $L^{(n)}$  est la représentation de  $S^{(n)}$  obtenue à l'aide de l'application  $g$ . Si la fonction d'affectation (permettant de passer de  $L^{(n)}$  à  $S^{(n+1)}$ ) et la fonction de représentation (permettant de passer de  $S^{(n+1)}$  à  $L^{(n+1)}$ ) sont bien choisies, on peut généralement montrer que les suites  $u_n$  et  $v_n$  sont convergentes et que la suite  $u_n$  est décroissante ; autrement dit, l'algorithme fait décroître le critère à chaque itération jusqu'à stabilisation.

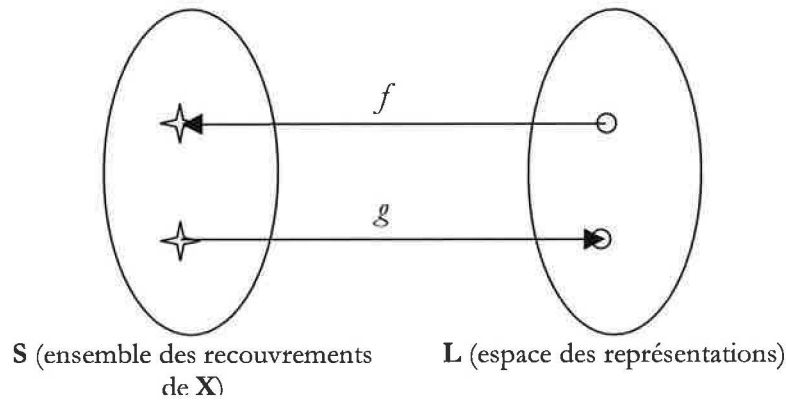


Figure 2

Soit  $L_k = L \times L \times \dots \times L$  ( $k$  fois) et  $P_k$  l'ensemble des partitions à  $k$  classes de  $X$  ( $P$  est l'ensemble des parties de  $X$ ). On considère  $D : P \times L \rightarrow \mathbb{R}$  vérifiant les hypothèses suivantes :

- le minimum de  $D(A, L)$  existe et est unique et ce  $\forall A \in P$  ;
- $D(A, l) \stackrel{\text{def}}{=} \sum_{a \in A} \Omega_a D(a, l) \quad \forall A \in P \text{ et } l \in L$ .

On définit le critère à optimiser  $\mathbf{W}$  l'application  $W : P_k \times L_k \rightarrow \mathbb{R}^+$  telle que :

$$W(P, L) \stackrel{\text{def}}{=} \sum_{i=1}^k D(P_i, L_i)$$

(ou encore  $W(P, L) \stackrel{\text{def}}{=} \sum_{i=1}^k \sum_{x \in P_i} D(x, L_i)$ ).

Nous rappelons que  $\mathbf{W}$  mesure l'adéquation entre une partition  $P$  de  $X$  et la représentation de ses classes  $L$ . On cherche donc parmi tous les éléments de  $(P, L) \in P_k \times L_k$  celui qui minimise  $\mathbf{W}$ .

La fonction de représentation  $g: P_k \rightarrow L_k$  est telle que  $g(P) = L$  avec  $P = (P_1, \dots, P_k)$  et  $L = (L_1, \dots, L_k)$  où  $L_i$  est un élément de  $L$  qui minimise  $D(P_i, \cdot)$ .

Quant à la fonction d'affectation, c'est l'application  $f: L_k \rightarrow P_k$  telle que  $f(L) = P$  où  $L = (L_1, \dots, L_k)$ ,  $P = (P_1, \dots, P_k)$  et  $P_i = \{x \in X / D(x, L_i) \leq D(x, L_j)\}$  ( $i < j$  en cas d'égalité).

L'algorithme se formalise alors de la façon suivante :

- solution initiale :  $v_0 = (P^{(0)}, L^{(0)})$  estimée ou tirée au hasard ;
- $v_n = (P^{(n)}, L^{(n)}) \in P_k \times L_k$  où  $L^{(n)} = g(P^{(n-1)})$  et  $P^{(n)} = f(g(P^{(n-1)}))$
- $u_n = W(v_n)$

Il est important de noter que l'algorithme converge vers une solution *non nécessairement unique*, ceci dépendant du tirage de départ.

Exemple :

Cas où l'algorithme des nuées dynamiques ne converge pas vers une solution unique :



Figure 3

Dans cet exemple, il y a plusieurs solutions optimales équivalentes. Celle trouvée par l'algorithme dépend de la solution initiale. Le triangle défini par les points est équilatéral. Notons qu'ici  $a = \frac{\sqrt{3}}{2}$  et que les ellipses traduisent simplement le regroupement des points dans une même classe.

Nous allons à présent démontrer successivement que  $u_n$  converge en décroissant et que  $v_n$  converge.

Proposition 1 :

$u_n$  converge en décroissant.

Démonstration (adaptée de E.Diday [13]) :

Nous allons montrer que  $u_{n-1} \geq W(P^{(n-1)}, L^{(n)}) \geq u_n$ .

La première inégalité est vraie car :

$\forall i \quad D(P_i^{(n-1)}, L_i^{(n-1)}) \geq D(P_i^{(n-1)}, L_i^{(n)})$  par définition de  $g$ , d'où :

$$u_{n-1} = W(P^{(n-1)}, L^{(n-1)}) \geq W(P^{(n-1)}, L^{(n)}).$$

Quant à la seconde inégalité :

$$\sum_{i=1}^k \sum_{x \in P_i^{(n-1)}} D(x, L_i^{(n)}) \geq \sum_{i=1}^k \sum_{x \in P_i^{(n)}} D(x, L_i^{(n)})$$

Elle est également vraie car tout élément  $x \in X$  est plus proche au sens de  $D$  de la représentation de sa classe dans la partition  $P^{(n)}$  que dans la partition  $P^{(n-1)}$ , ceci par définition même de l'application  $f$ .

La suite  $u_n$  étant décroissante et minorée (car positive), elle converge. ■

Proposition 2 :

La suite  $v_n$  converge.

Démonstration (adaptée de E.Diday [13]) :

On sait que toute suite convergente sur un ensemble fini atteint sa limite, or c'est le cas de la suite  $u_n$  puisque l'espace est fini.

Supposons donc que la convergence soit atteinte à l'itération  $N$ , autrement dit que  $u_{N+1} = u_N$  d'où  $W(V_{N+1}) = W(V_N)$  ; cette égalité n'implique pas forcément que  $v_{N+1} = v_N$  car  $W$  n'est pas nécessairement injective. Nous allons néanmoins montrer que  $v_n = v_N \quad \forall n \geq N$ .

En effet,  $u_{N+1} = u_N \Rightarrow W(V_{N+1}) = W(V_N)$ , or :

$$\left. \begin{array}{l} W(L^{(N)}, P^{(N)}) = W(L^{(N+1)}, P^{(N+1)}) \\ g(P^{(N)}) = L^{(N+1)} \end{array} \right\} \Rightarrow L^{(N)} = L^{(N+1)}$$

car  $L^{(N)} \neq L^{(N+1)}$  impose  $D(P_i^{(N)}, L_i^{(N)}) > D(P_i^{(N)}, L_i^{(N+1)})$  si  $L_i^N \neq L_i^{N+1}$  (et il existe nécessairement  $i$  tel que ce soit le cas). L'égalité est obtenue si  $L_i^N = L_i^{N+1}$ .

D'où en sommant pour  $i = \{1, \dots, k\}$  :

$$W(L^{(N)}, P^{(N)}) > W(L^{(N+1)}, P^{(N)}).$$

D'autre part,  $W(L^{(N+1)}, P^{(N)}) > W(L^{(N+1)}, P^{(N+1)})$  (voir Proposition 1).

D'où :

$W(L^{(N)}, P^{(N)}) > W(L^{(N+1)}, P^{(N+1)})$ , ce qui est contraire à l'hypothèse  $u_{N+1} = u_N$ . On a donc  $L^{(N)} = L^{(N+1)}$ . Par conséquent :

$f(L^{(N)}) = f(L^{(N+1)})$  donc  $P^{(N)} = P^{(N+1)} + 1$  et finalement  $v_N = v_{N+1}$ .

On peut alors recommencer le même raisonnement pour tout  $n \geq N$ , d'où

$$v_n = v_N \quad \forall n \geq N \quad \blacksquare$$

La solution obtenue après exécution de l'algorithme présente les propriétés suivantes :

- elle est « stable » et « non-biaisée » car la représentation  $L^* = (L_1^*, \dots, L_k^*)$  obtenue à la convergence pour les  $k$  classes induit (par  $f$ ) une partition  $P^*$  dont les  $k$  classes admettent justement pour représentation  $L^*$  ;
- la représentation obtenue  $L^*$  permet la construction d'une fonction d'affectation. Cette fonction associe à chaque élément de  $X$  la classe d'indice  $i$  telle que  $D(x, L_i^*) = \min(D(x, L_j^*))$  ;
- on peut obtenir plusieurs solutions en changeant le choix de la partition  $P^{(0)}$  faite au départ. Si une seule solution doit être retenue on choisira naturellement celle qui donne la meilleure valeur au critère  $W$ .

## 4. CONSTRUCTION DU E-SQUELETTE.

Ce paragraphe décrit la manière dont nous avons spécialisé chacun des maillons des nuées dynamiques. Elle constitue surtout un exemple, et ne doit en aucune façon être considérée comme exclusive.

### 4.1. Considérations à propos de l'espace à partitionner.

Considérons un solide  $A$ . Son discrétisé [10], noté  $\tilde{A}$ , est défini par un ensemble fini de  $m$  éléments de volume (*voxels*)  $v_i$ , dans le cadre d'une fréquence d'échantillonnage constante. Chacun de ces  $v_i$  est un parallélépipède rectangle dont les coordonnées des sommets sont liées d'une part à la résolution de la discrétisation, et d'autre part à l'épaisseur et la position spatiale dans le cas d'une acquisition par coupes tomodynamométriques. On caractérise alors chaque voxel par le quadruplet :

$$p_i(x_i, y_i, z_i, \omega_i)$$

où  $\omega_i$  est une unité de masse ou de volume vérifiant la relation :

$$\sum_{i=1}^m \omega_i = 1$$

Équation 7

Il apparaît que  $\tilde{A}$  est le *nuage probabilisé*  $N(X, Y, Z, \omega)$  des  $m$  points par rapport au système d'axes fixes  $xyz$ , d'origine  $O$ . Chaque point est doté d'une pondération  $\omega_i$ .

### 4.2. La fonction de représentation.

#### 4.2.1. Définition de la matrice de dispersion.

Comme nous le verrons plus loin, le paramètre fondamental caractérisant chaque classe est la *matrice de dispersion*. En effet, elle offre une représentation synthétique de l'ensemble des voxels puisqu'elle quantifie sa *dispersion géométrique* : la forme, l'étendue et la distribution de la masse est décrite relativement à des axes pertinents et intuitifs appelés *vecteurs propres*. Sous certaines hypothèses que nous énoncerons, la matrice de dispersion est assimilable à la *matrice d'inertie*, et ses vecteurs propres sont alors les *axes principaux d'inertie*. L'atout majeur de ce type de paramètre est son indépendance vis-à-vis non seulement de la résolution mais aussi de l'orientation et du positionnement de l'objet dans l'espace. Nous allons maintenant présenter la théorie issue de la mécanique des solides [4] et démontrer les propriétés énoncées dans ce paragraphe.



La *matrice de dispersion* traduit la dispersion des points du nuage  $N(X, Y, Z, \omega)$  :

$$M_{\tilde{N}}^D = \begin{bmatrix} \text{var}(X) & \text{cov}(Y, X) & \text{cov}(Z, X) \\ \text{cov}(X, Y) & \text{var}(Y) & \text{cov}(Z, Y) \\ \text{cov}(X, Z) & \text{cov}(Y, Z) & \text{var}(Z) \end{bmatrix}$$

où les termes  $\text{var}$  et  $\text{cov}$  désignent respectivement la variance et la *covariance*, définies par :

$$\text{var}(X) = \sum_{i=1}^m (x_i - \bar{x})^2 \omega_i \quad \text{var}(Y) = \sum_{i=1}^m (y_i - \bar{y})^2 \omega_i \quad \text{var}(Z) = \sum_{i=1}^m (z_i - \bar{z})^2 \omega_i$$

Équation 8

$$\begin{aligned} \text{cov}(X, Y) &= \sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y}) \omega_i = \text{cov}(Y, X) \\ \text{cov}(Y, Z) &= \sum_{i=1}^m (y_i - \bar{y})(z_i - \bar{z}) \omega_i = \text{cov}(Z, Y) \\ \text{cov}(X, Z) &= \sum_{i=1}^m (x_i - \bar{x})(z_i - \bar{z}) \omega_i = \text{cov}(Z, X) \end{aligned}$$

Équation 9

où  $\bar{x}$ ,  $\bar{y}$  et  $\bar{z}$  sont les *moyennes* ou *espérances mathématiques* des variables concernées :

$$\bar{x} = \sum_{i=1}^m x_i \omega_i \quad \bar{y} = \sum_{i=1}^m y_i \omega_i \quad \bar{z} = \sum_{i=1}^m z_i \omega_i$$

Équation 10

Rappelons que  $\sum_{i=1}^m \omega_i = 1$ .

Nous remarquerons que  $M_{\tilde{N}}^D$  est symétrique du fait des propriétés de la covariance.

Nous caractérisons la dispersion des points du nuage  $N(X, Y, Z, \omega)$  par les *composantes principales* du nuage obtenu à partir de la matrice de dispersion. C'est un ensemble de trois axes perpendiculaires deux à deux formant une base dans laquelle la matrice de dispersion est *diagonale*.

Une matrice est diagonale si et seulement si elle est exprimée dans un système d'axes formé par ses vecteurs propres. Les éléments de la diagonale sont alors les *valeurs propres* correspondantes. Les directions des composantes principales du nuage de points sont donc définies par les vecteurs propres  $v$  de la matrice de dispersion  $M_{\tilde{N}}^D$  pour lesquels la relation suivante est vraie :

$$M_N^D(\tilde{v}_\lambda) = \lambda \cdot \tilde{v}_\lambda$$

Équation 11

où  $\lambda$  est un scalaire. On dira que le vecteur  $\tilde{v}_\lambda$  de coordonnées  $(\tilde{v}_{\lambda,X}, \tilde{v}_{\lambda,Y}, \tilde{v}_{\lambda,Z})$  est le *vecteur propre* de  $M_N^D$  correspondant à la *valeur propre*  $\lambda$ .

L'Équation 11 produit le système :

$$\begin{cases} (\text{var}(X) - \lambda) \cdot \tilde{v}_{\lambda,X} + \text{cov}(Y, X) \cdot \tilde{v}_{\lambda,Y} + \text{cov}(Z, X) \cdot \tilde{v}_{\lambda,Z} = 0 \\ \text{cov}(X, Y) \cdot \tilde{v}_{\lambda,X} + (\text{var}(Y) - \lambda) \cdot \tilde{v}_{\lambda,Y} + \text{cov}(Z, Y) \cdot \tilde{v}_{\lambda,Z} = 0 \\ \text{cov}(X, Z) \cdot \tilde{v}_{\lambda,X} + \text{cov}(Y, Z) \cdot \tilde{v}_{\lambda,Y} + (\text{var}(Z) - \lambda) \cdot \tilde{v}_{\lambda,Z} = 0 \end{cases}$$

Équation 12

Pour que l'Équation 12 ait des solutions non triviales i.e. non nulles, il faut que :

$$\begin{vmatrix} \text{var}(X) - \lambda & \text{cov}(Y, X) & \text{cov}(Z, X) \\ \text{cov}(X, Y) & \text{var}(Y) - \lambda & \text{cov}(Z, Y) \\ \text{cov}(X, Z) & \text{cov}(Y, Z) & \text{var}(Z) - \lambda \end{vmatrix} = 0$$

Équation 13

L'Équation 13 conduit alors à une équation du troisième degré en  $\lambda$  qui admet trois racines réelles  $\lambda_1, \lambda_2$  et  $\lambda_3$ . L'Équation 10 détermine la direction des composantes principales, en déterminant pour chaque valeur propre  $\lambda_i$  les coordonnées  $(\tilde{v}_{\lambda_i,X}, \tilde{v}_{\lambda_i,Y}, \tilde{v}_{\lambda_i,Z})$  du vecteur propre associé  $\tilde{v}_{\lambda_i}$ .

Nous allons à présent montrer plusieurs propriétés liées à la matrice de dispersion, à savoir :

- la matrice de dispersion est invariante par translation et duplication (sur-échantillonnage constant) du nuage de points ;
- la matrice de dispersion est invariante par sur-échantillonnage du nuage de points ;
- sous certaines conditions, composantes principales et axes principaux d'inertie sont identiques.

Propriété 3 :

La matrice de dispersion  $M_{\tilde{N}}^D$  est invariante par translation du nuage de points  $\tilde{N}(X,Y,Z,\omega)$ .

Démonstration :

Soit  $\tilde{N}(X',Y',Z',\omega')$  le translaté du nuage probabilisé  $\tilde{N}(X,Y,Z,\omega)$  par le vecteur  $(\delta_x, \delta_y, \delta_z)$ . On se propose de démontrer la relation suivante :

$$M_{\tilde{N}(X,Y,Z,\omega)}^D = M_{\tilde{N}(X',Y',Z',\omega')}^D$$

Équation 14

$\forall i \in \{1, \dots, m\}$  on pose :

$$\begin{cases} x'_i = x_i + \delta_x \\ y'_i = y_i + \delta_y \\ z'_i = z_i + \delta_z \end{cases} \text{ et } \omega'_i = \omega_i$$

Par l'Équation 10 on obtient  $\begin{cases} \bar{x}' = \bar{x} + \delta_x \\ \bar{y}' = \bar{y} + \delta_y \\ \bar{z}' = \bar{z} + \delta_z \end{cases}$ , ce qui conduit à  $\begin{cases} x'_i - \bar{x}' = x_i - \bar{x} \\ y'_i - \bar{y}' = y_i - \bar{y} \\ z'_i - \bar{z}' = z_i - \bar{z} \end{cases}$ .

En utilisant l'Équation 8 et l'Équation 9, on trouve :

$$\begin{aligned} \text{var}(X) &= \text{var}(X'), \text{var}(Y) = \text{var}(Y'), \text{var}(Z) = \text{var}(Z') \\ \text{cov}(X, Y) &= \text{cov}(X', Y'), \dots \end{aligned}$$

L'Équation 14 est donc vérifiée. ■

Propriété 4 :

La matrice de dispersion  $M_{\tilde{N}}^D$  est invariante par duplication du nuage de points  $\tilde{N}(X,Y,Z,\omega)$ .

Démonstration (adaptée de P. Borianne [10]) :

Soit  $\tilde{N}(\alpha X, \alpha Y, \alpha Z, \omega)$  où  $\alpha$  est le facteur de duplication. On note  $(x'_{i,j}, y'_{i,j}, z'_{i,j})$  les coordonnées des  $\alpha m$  points obtenus en dupliquant  $\alpha$  fois les  $m$  points  $(x_i, y_i, z_i)$ . Les points  $(x'_{i,j}, y'_{i,j}, z'_{i,j})$  constituent le nuage probabilisé  $\tilde{N}(X', Y', Z', \omega')$ . On se propose de démontrer la relation suivante :

$$M_{N(X,Y,Z,\omega)}^D = M_{N(X',Y',Z',\omega')}^D$$

Équation 15

$\forall i \in \{1, \dots, m\}$  et  $\forall j \in \{1, \dots, \alpha\}$  on pose :

$$\begin{cases} x'_{i,j} = x_i \\ y'_{i,j} = y_i \text{ et } \omega'_{i,j} = \frac{\omega_i}{\alpha} \\ z'_{i,j} = z_i \end{cases}$$

Avec l'Équation 10 on trouve  $\begin{cases} \bar{x}' = \bar{x} \\ \bar{y}' = \bar{y} \\ \bar{z}' = \bar{z} \end{cases}$  et l'on montre alors aisément que :

$$\begin{cases} \sum_{j=1}^{\alpha} (x'_{i,j} - \bar{x}')^2 \omega'_{i,j} = (x_i - \bar{x})^2 \omega_i \\ \sum_{j=1}^{\alpha} (y'_{i,j} - \bar{y}')^2 \omega'_{i,j} = (y_i - \bar{y})^2 \omega_i \\ \sum_{j=1}^{\alpha} (z'_{i,j} - \bar{z}')^2 \omega'_{i,j} = (z_i - \bar{z})^2 \omega_i \end{cases}$$

et que :

$$\begin{cases} \sum_{j=1}^{\alpha} (x'_{i,j} - \bar{x}')(y'_{i,j} - \bar{y}')(z'_{i,j} - \bar{z}') = (x_i - \bar{x})(y_i - \bar{y})(z_i - \bar{z}) \\ \dots \\ \dots \end{cases}$$

d'où, en employant l'Équation 8 et l'Équation 9 :

$$\begin{aligned} \text{var}(X) &= \text{var}(X'), \text{var}(Y) = \text{var}(Y'), \text{var}(Z) = \text{var}(Z') \\ \text{cov}(X, Y) &= \text{cov}(X', Y'), \dots \end{aligned}$$

La relation donnée dans l'Équation 15 est donc vraie. ■

Propriété 5 :

La matrice de dispersion  $M_{\tilde{N}}^D$  est invariante par suréchantillonnage du nuage de points  $\tilde{N}(X, Y, Z, \omega)$ .

Démonstration (adaptée de P. Borianne [10]) :

On désigne le suréchantillonnage de facteur  $\alpha$  suivant une direction  $z$  du nuage probabilisé  $\tilde{N}(X, Y, Z, \omega)$ , l'ensemble  $\tilde{N}(X'', Y'', Z'', \omega'')$  des  $\alpha m$  points obtenu par superposition des  $\alpha$  nuages  $\tilde{N}(X', Y', Z', \omega')$  obtenus par translations successives et constantes du nuage initial  $\tilde{N}(X, Y, Z, \omega)$ . Il s'agit en fait, dans le cas d'une numérisation par coupes d'épaisseur  $e$ , de les subdiviser en coupes d'épaisseur  $\frac{e}{\alpha}$ .

$\forall i \in \{1, \dots, m\}$  et  $\forall j \in \{1, \dots, \alpha\}$  on pose :

$$\begin{cases} x'_{i,j} = x_i \\ y'_{i,j} = y_i \\ z'_{i,j} = z_i + \delta_{z_j} \end{cases} \text{ et } \omega'_{i,j} = \frac{\omega_i}{\alpha}, \delta_{z_j} = j\delta_z \text{ avec } \delta_z = \frac{e}{\alpha}$$

La matrice de dispersion est alors définie par :

$$M_{\tilde{N}(X'', Y'', Z'', \omega'')}^D = M_{\sum_{j=1}^{\alpha} \tilde{N}(X, Y, Z + \delta_{z_j}, \omega)}^D$$

qui devient, d'après l'Équation 14 et l'Équation 15 :

$$M_{\tilde{N}(X'', Y'', Z'', \omega'')}^D = M_{\tilde{N}(X, Y, Z, \omega)}^D$$

■

Propriété 6 :

Sous certaines conditions que nous allons énoncer dans la démonstration, composantes principales et axes principaux d'inertie sont rigoureusement identiques.

Démonstration (adaptée de P. Borianne [10]) :

La matrice de dispersion s'exprime en fonction du *tenseur des moments d'inertie*. Ces deux formes matricielles admettent les mêmes vecteurs propres et leurs valeurs propres respectives sont identiques à une translation près.

Nous rappelons l'expression du tenseur des moments d'inertie d'un nuage de  $m$  points matériels de masse  $\omega_i$  et de coordonnées  $(x_i, y_i, z_i)$  par rapport au système d'axes fixes  $xyz$  d'origine  $O$  :

$$T_{\tilde{N}(X,Y,Z,\omega)} = \begin{bmatrix} I(X,X) & I(Y,X) & I(Z,X) \\ I(X,Y) & I(Y,Y) & I(Z,Y) \\ I(X,Z) & I(Y,Z) & I(Z,Z) \end{bmatrix}$$

Équation 16

où les quantités  $I(X,X)$ ,  $I(Y,Y)$  et  $I(Z,Z)$  sont appelées *moments d'inertie* et sont définies par :

$$I(X,X) = \sum_{i=1}^m \omega_i (y_i^2 + z_i^2), \quad I(Y,Y) = \sum_{i=1}^m \omega_i (x_i^2 + z_i^2), \quad I(Z,Z) = \sum_{i=1}^m \omega_i (x_i^2 + y_i^2)$$

Équation 17

Et les quantités  $I(X,Y)$ ,  $I(X,Z)$ , ... sont nommées *produits d'inertie* et sont définies ainsi :

$$I(X,Y) = -\sum_{i=1}^m \omega_i x_i y_i = I(Y,X), \quad I(Y,Z) = -\sum_{i=1}^m \omega_i y_i z_i = I(Z,Y), \\ I(X,Z) = -\sum_{i=1}^m \omega_i x_i z_i = I(Z,X)$$

Équation 18

Si  $O$  est le centre de gravité du nuage de points, l'expression de la matrice d'inertie est simplifiée puisque  $\bar{x} = \bar{y} = \bar{z} = 0$ , nous obtenons alors l'équivalence avec la matrice de dispersion de la manière suivante :

$$T_{\tilde{N}(X,Y,Z,\omega)} = \eta \cdot \text{Id} - M_{\tilde{N}(X,Y,Z,\omega)}^D$$

Équation 19

où  $\text{Id}$  est la matrice identité et  $\eta = \sum_{i=1}^m \omega_i (x_i^2 + y_i^2 + z_i^2)$ .

Montrons à présent que tout vecteur propre du tenseur des moments d'inertie est également vecteur propre de la matrice de dispersion (et réciproquement).

Soit  $\tilde{v}_\lambda$  le vecteur propre du tenseur  $T_{\tilde{N}(X,Y,Z,\omega)}$  correspondant à la valeur propre  $\lambda$ . On a par définition :

$$T_{\tilde{N}(X,Y,Z,\omega)}(\tilde{v}_\lambda) = \lambda \tilde{v}_\lambda$$

D'après l'Équation 19, cette relation s'écrit encore comme :

$$\mathbf{M}_{\mathbf{N}(X,Y,Z,\omega)}^{\mathbf{D}}(\tilde{\mathbf{v}}_{\lambda}) = (\eta - \lambda)\tilde{\mathbf{v}}_{\lambda}$$

$\tilde{\mathbf{v}}_{\lambda}$  est donc le vecteur propre de la matrice de dispersion  $\mathbf{M}_{\mathbf{N}(X,Y,Z,\omega)}^{\mathbf{D}}$  correspondant à la valeur propre  $(\eta - \lambda)$ . Réciproquement, on montre que tout vecteur propre de la matrice de dispersion est également vecteur propre du tenseur des moments d'inertie.

Les valeurs propres  $\lambda_T^1, \lambda_T^2, \lambda_T^3$  du tenseur des moments d'inertie sont liées aux valeurs propres  $\lambda_M^1, \lambda_M^2, \lambda_M^3$  de la matrice de dispersion par la relation :

$$\lambda_M^p = \eta - \lambda_T^p \text{ avec } p \in \{1, 2, 3\}$$

Équation 20

Selon l'Équation 20, nous obtenons l'équivalence de la matrice de dispersion et de la matrice d'inertie, à la condition que les points soient exprimés relativement au centre de gravité local à l'objet. ■

#### 4.2.2. Choix du représentant de classe.

L'utilisation de la matrice de dispersion (ou d'inertie) nous est apparue intéressante [10] pour plusieurs raisons :

- la matrice de dispersion présente de bonnes propriétés de caractérisation [5] (voir section précédente 4.2.1) ;
- les grandeurs extraites de cette analyse sont très intuitives puisque basées sur des propriétés mécaniques ;
- les cas dégénérés (classes isotropes, ou planes) sont facilement décelables en observant le rang de la matrice de dispersion, et n'entraînent pas d'erreurs ou de sortie du domaine de définition de la méthode de calcul.

Nous proposons donc comme représentant de classe le couple formé de la matrice de dispersion (nous verrons pourquoi en section 4.6.2) de la classe ainsi que la position du centre de gravité de la classe. À des fins de normalisation, tout vecteur principal d'inertie et toute coordonnée de centre de gravité sont exprimés dans le repère local à l'objet (d'origine le centre de gravité de l'objet entier et d'axes les vecteurs principaux d'inertie de l'objet entier).

L'aspect modulaire de la fonction de représentation autorise la possibilité d'intégrer de nouvelles fonctions de caractérisation, qui sera abordée dans la section *Reconnaissance de Formes*.

### **4.3. La fonction de distance.**

Etroitement liée à la fonction de représentation, elle est l'un des paramètres fondamentaux de la méthode des nuées dynamiques. Nous allons découvrir quels sont les outils mathématiques disponibles et expliquer les choix effectués.

#### **4.3.1. Les métriques.**

Différentes métriques peuvent être introduites dans l'algorithme, qui vont conditionner le partitionnement [17][22][23]. La distance euclidienne exhibe une grande stabilité et une faible potentialité de bouclage des nuées dynamiques (oscillations infinies de points d'une classe à l'autre à chaque itération). Les classes ainsi produites convergent vers des dispersions sphériques, i.e. la matrice de dispersion est de rang 1 après un grand nombre d'itérations. Ce phénomène semble intéressant puisqu'il permet une très bonne approximation géométrique (nous le verrons plus tard) pour la reconstruction à un niveau d'itération important de l'algorithme. Les premières classes restent néanmoins fortement caractéristiques au sens mécanique, à savoir que le rang de la matrice de dispersion reste égal à trois.

La distance de Mahalanobis [16] (voir chapitre tour d'horizon en reconnaissance de formes) nous a aussi semblé intéressante puisqu'elle normalise les facteurs d'homothéties des classes selon les axes d'inertie de celles-ci. De cette manière, les classes « allongées » sont parfaitement détectées et caractérisées. Pour ce faire, nous avons introduit la matrice de dispersion dans l'expression de la distance de Mahalanobis. Employer une telle distance revient à exécuter une transformation affine sur la classe, ce qui la rend sphérique et réduit de ce fait les pertes de points des classes allongées (les points éloignés du centre de gravité initial sont plus susceptibles de changer de classe, ce qui produit les phénomènes de chapelet).

Nous nous sommes toutefois heurtés à de nombreux problèmes :

- la matrice de dispersion peut être singulière (peu d'échantillons, aucune anisotropie...etc.), ce qui rend alors le calcul de la matrice inverse impossible. Ce problème peut toutefois être contourné en empêchant le découpage d'une classe sphérique – avec toutes les limitations que cela sous-entend ;
- son intégration aux nuées dynamiques produit des classes parfois très grandes ou très petites selon la dispersion : si celle-ci est très grande, les covariances le sont d'autant plus, ce qui produit alors de petites valeurs de distance de Mahalanobis. Un phénomène de « rattachement massif » de points à une classe survient dans ce cas ;
- elle n'exhibe pas un comportement de décroissance monotone à cause de l'emploi de la matrice de dispersion ;
- elle produit de fréquents phénomènes de bouclage infini des nuées dynamiques.



Les défauts de classes très grandes ou très petites peuvent en partie être corrigés par l'emploi de la *distance de Mahalanobis pondérée*, au prix d'une perte de robustesse et de généralité. Le manque d'intuitivité des pondérations nous a également semblé inacceptable.

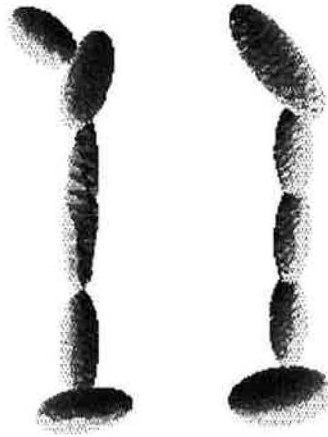
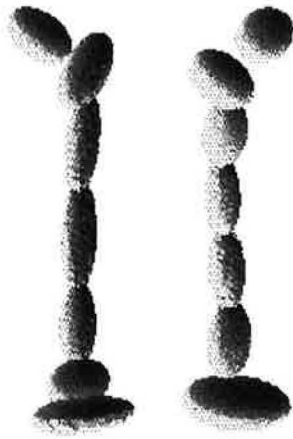


Figure 4

A gauche : partitionnement avec la distance de Mahalanobis. A droite : partitionnement avec la distance Euclidienne. (E-Squelette de niveau sémantique 4). Le thème de la représentation graphique des objets sera traité au chapitre suivant (chapitre 6).

Dans la Figure 4, la distance de Mahalanobis semble mettre en exergue les classes allongées, ce qui est ici très pertinent pour la partie médiane de l'objet (fémur) qui est tubulaire.



Pour sept classes, le phénomène de « chapelet » est présent à droite (distance euclidienne). Il est moins marqué pour la distance de Mahalanobis (à gauche), mais un problème survient en bas de l'objet : la classe devient tassée. Comme expliqué précédemment, la classe voisine de celle-ci possède une forte covariance et devient un attracteur de points. A ce moment là, la caractérisation géométrique de la classe inférieure est perdue, ce qui est confirmé lorsque nous passons au niveau sémantique supérieur (image de droite). Le chapelet commence à apparaître, ce qui n'apporte finalement pas de réponse intéressante au

problème, et ne justifie donc pas le surcoût en temps de calcul.



L'emploi d'une distance pondérée de Mahalanobis est possible, mais se fera au prix d'une perte de robustesse et d'intuitivité de mesure. Plus généralement, ce type de distance ouvre la voie aux *distances adaptatives*, qui varient en fonction du contexte (forme de la classe, densité...Etc.). L'aspect arbitraire des pondérations et des heuristiques de régulation nous a semblé constituer un obstacle majeur à l'utilisation de telles techniques [17].

#### 4.3.2. Choix de la fonction de distance D .

La distance euclidienne exhibe un très bon comportement en termes de robustesse et de coût calculatoire. Elle produit cependant un phénomène de « chapelet », c'est-à-dire qu'elle guide l'algorithme de partitionnement vers un morcellement des classes allongées. Les classes convergent vers une structure sphérique puisqu'aucune pondération n'induit d'anisotropie. Ce dernier point n'est pas problématique en soi puisqu'il survient pour un nombre « assez grand » de classe, et convient parfaitement à l'approximation géométrique.

#### 4.4. Le critère d'optimisation W.

Ce critère combine la fonction de représentation de chaque classe – ici les centres de gravité – avec la fonction de distance.

A chaque itération, le critère à minimiser est la distance entre les points et les centres de gravité de chaque classe présente, autrement dit :

$$W(P, L) \stackrel{\text{def}}{=} \sum_{i=1}^k D(P_i, L_i)$$

Équation 21

Où  $k$  est le nombre de classes,  $P_i$  et  $L_i$  les noyaux de chaque classe (ici les centres de gravité).

Le critère dépend bien de la fonction de distance qui est définie comme suit :

$$D(P_i, L_i) = \sum_{x \in P_i} \omega_i d^2(x, L_i)$$

Équation 22

où  $d$  est la distance euclidienne classique.

#### 4.5. Minimisation du critère d'optimisation W.

Pour minimiser W, nous procédons classiquement comme suit :

Répéter

Pour Chaque Classe Existante  $P_i$  Faire

    Calculer son centre de gravité  $G_i$

FinPour

Pour Chaque  $x \in \tilde{N}$  Faire

    Rattacher  $x$  à la classe  $P_i$  dont le centre de gravité  
 est le plus proche au sens de  $d$

FinPour

Tant Que les  $G_i$  Changent.

Algorithme 1

#### Remarque :

Bien que théoriquement impossible, le phénomène de bouclage dans la section « Répéter ... TantQue les  $G_i$  changent » peut survenir à cause d'une imprécision numérique survenant lors du calcul des distances. Les heuristiques habituelles consistent à stopper le processus lorsque le nombre de points changeant de classe au cours des nuées dynamiques est inférieur à un certain seuil.

#### 4.6. L'Algorithme *progressif* de construction du E-Squelette.

Maintenant que nous possédons les éléments clés pour exécuter les nuées dynamiques, nous allons décrire l'algorithme global de construction du E-Squelette, nommé *algorithme progressif*.

L'algorithme progressif de partitionnement propose une technique originale et automatique, avec des paramètres intuitifs à mettre en œuvre. Les seuils, souvent présents dans ce type de méthodes, ont été supprimés de manière à améliorer la robustesse et la simplicité d'utilisation.

Il se déroule de la manière suivante (méthode classique) :

```
Tous les points appartiennent à la même classe
Tant Que NombreDeClasses < NombreDeClassesVoulu

Pour Chaque classe  $P_i$ 
    Calculer la Matrice d'Inertie  $M_{P_i}^D$ 
    Découper virtuellement  $P_i$  en Deux Sous-Classes  $P_i'$  et  $P_i''$ 
    Opérer les Nuées Dynamiques sur toutes les Classes
    Evaluer le Critère de Scission  $r_i$ 
Fin Pour

Découper la classe  $P_i$  de Critère  $r_i$  Minimal
Opérer les Nuées Dynamiques sur toutes les Classes

Fin Tant Que
```

Algorithme 2

##### 4.6.1. La solution initiale, point de départ des nuées dynamiques.

Pour initialiser l'algorithme, c'est la matrice de dispersion et le centre de gravité de l'objet entier qui sont utilisés. Rappelons ici que ce sont *tous* les points décrivant l'objet qui sont employés, et pas seulement les points de contour.

Le trièdre de vecteurs propres de la matrice d'inertie ainsi que le centre de gravité constituent la base locale de l'objet dans laquelle pourront être exprimés tous les autres vecteurs.

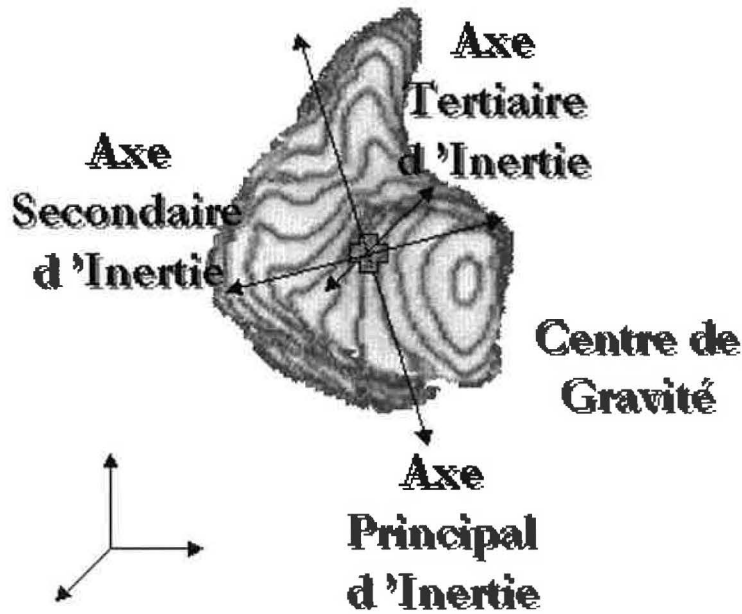


Figure 5

La classe initiale est constituée du nuage entier. Le nuage est caractérisé par les données extraites de la matrice de dispersion des points, assimilable à la matrice d'inertie dans ce cas.

#### 4.6.2. Découpage d'une classe en deux sous-classes.

Cette étape est cruciale puisque du fait des propriétés des nuées dynamiques, elle influe directement sur la vitesse de convergence. Il faut donc proposer une solution initiale qui soit "la plus proche possible" de la solution idéale. Pour ce faire nous proposons le découpage suivant :

Soit  $\kappa_i'$  le vecteur propre associé à la plus grande valeur propre (celles-ci sont toutes positives) de la matrice de dispersion  $M_{P_i}^D$  de la classe  $P_i$ . Soit  $m_i$  le nombre de points  $x_k^i$  vérifiant  $x_k^i \in P_i$ . On calcule alors la valeur suivante :

$$\xi_i = \frac{1}{m_i} \sum_{k=1}^{m_i} \kappa_k' x_k^i$$

$\xi_i$  est la moyenne des produits scalaires des points de la classe  $P_i$  par le vecteur principal d'inertie  $\kappa_i'$ .

La classe  $P_i$  est partitionnée en deux sous-classes  $P_i'$  et  $P_i''$  vérifiant :

$$x_k^i \in P_i' \Leftrightarrow \kappa_k' \cdot x_k^i > \xi_i$$

$$x_k^i \in P_i'' \Leftrightarrow \kappa_k' \cdot x_k^i \leq \xi_i$$

En d'autres termes il s'agit de scinder le nuage de points considéré relativement au plan orthogonal à son axe principal d'inertie et passant par son centre de gravité. Nous espérons de cette manière fournir une solution initiale qui ne soit pas trop différente du partitionnement idéal (au sens du critère d'optimisation) et ainsi accélérer la méthode.

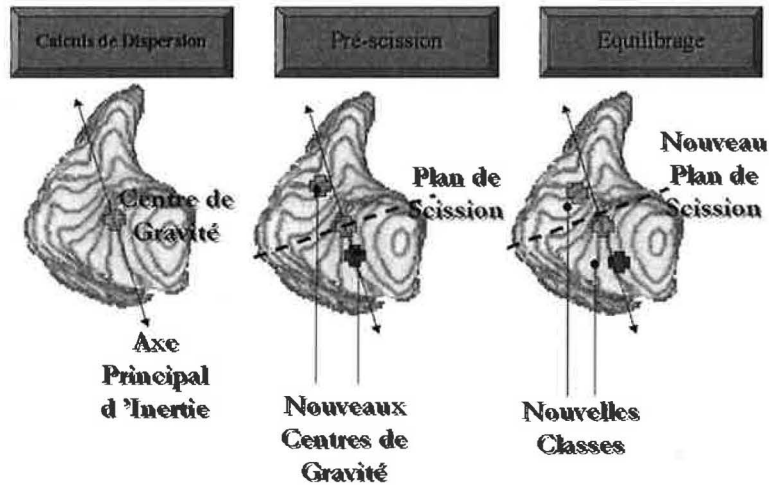


Figure 6

#### 4.6.3. Le critère de scission.

C'est une valeur réelle notée  $r_i$  qui décide de la meilleure classe à découper. Ce critère conditionne la vitesse de convergence du découpage, c'est-à-dire sa capacité de différenciation géométrique des sous-parties d'un objet.

Nous proposons le critère suivant :

$$r_i = \frac{V_i' + V_i''}{V_i}$$

Équation 23

où  $V_i$  est la *variance euclidienne* en  $x, y$ , et  $z$  de la classe  $P_i$ , et  $V_i', V_i''$  les variances euclidiennes des classes  $P_i'$  et  $P_i''$ . Soit  $\Omega_i$  le poids de la classe  $P_i$  et  $X_i, Y_i, Z_i$  les coordonnées en  $x, y, z$  des points de  $P_i$ ,  $V_i$  se définit alors comme :

$$V_i \stackrel{def}{=} \text{var}(\Omega_i, X_i) + \text{var}(\Omega_i, Y_i) + \text{var}(\Omega_i, Z_i)$$

Équation 24

Ces variances sont indépendantes du repère dans lequel sont exprimés les points. En effet, la propriété suivante est directe :

$$V_i^{def} = \text{var}(\Omega_i, X_i) + \text{var}(\Omega_i, Y_i) + \text{var}(\Omega_i, Z_i) = E(\Omega_i, (X - E(\Omega_i, X))^2 + (Y - E(\Omega_i, Y))^2 + (Z - E(\Omega_i, Z))^2)$$

Équation 25

$E$  désignant l'espérance de la variable associée. Les distances étant invariantes par translation, rotation et symétrie, il en est de même de l'espérance et donc de la variance euclidienne  $V_i$  qui reste indépendante du repère choisi. ■

Le critère  $r_i$  fournit ainsi une information normalisée sur la dispersion des points avant et après équilibrage par les nuées dynamiques, ce qui revient à supprimer l'influence de la taille des classes au profit de leurs discontinuités géométriques (pour une fréquence d'échantillonnage constante).

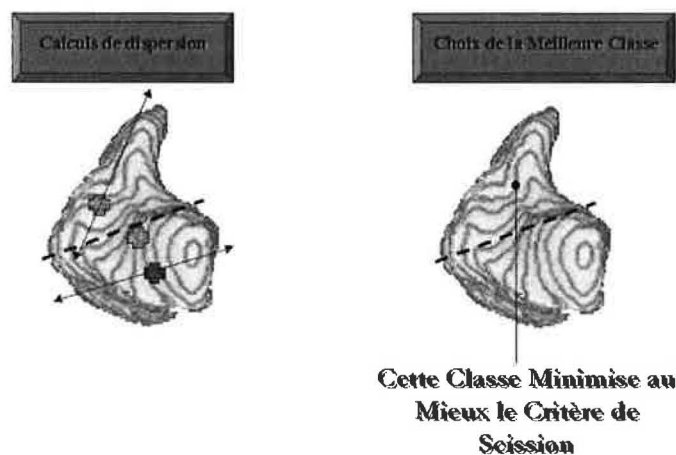


Figure 7

Un critère naïf de scission, qui a été mis en place en phase préparatoire, consistait à scinder la classe minimisant la variance intraclasse totale. Le scénario suivant explique pourquoi un tel critère n'est pas adapté :

Soit un nuage de points représentant trois entités ellipsoïdales, respectivement deux petites très proches et une grande. La fréquence d'échantillonnage est considérée comme étant constante et la dispersion des points dans l'espace comme étant homogène. Si nous décidons de couper la classe minimisant la variance intraclasse totale, alors c'est la grande classe qui sera découpée. Ce comportement n'est pas valable au sens de la caractérisation du nuage car il introduit un biais dans la configuration initiale des nuées dynamiques. Il est certes possible que celles-ci « corrigent » ultérieurement la situation mais il faut, autant que faire se peut, garantir une solution initiale la meilleure possible pour accélérer les nuées dynamiques (voir section 3.1.2).

Du fait du choix du critère, une classe est donc « détectée » selon les critères intuitifs suivants :

- une forme peu isotrope – forte concavité par exemple – produit un  $r_i$  très faible, et met donc la classe en position prioritaire dans le processus de partitionnement ;
- la position est ensuite précisée en fonction du volume ou de la masse de la classe en question. Plus celle-ci sera importante et plus elle sera différenciée tôt.



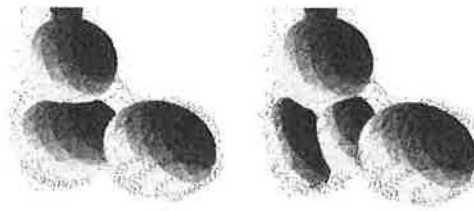


Figure 8

Détail de la scission d'une classe suivi d'un rééquilibrage : les classes voisines restent quasiment inchangées dans cet exemple, ce qui indique leur adéquation locale au nuage en terme de géométrie. Chaque ellipsoïde est une représentation graphique de la classe détectée.

#### **4.7. Propriétés de l'algorithme progressif.**

La propriété fondamentale de l'algorithme est qu'il ne dépend pas de l'ordre de traitement des points, condition *sine qua non* pour assurer l'unicité de décomposition à travers plusieurs acquisitions d'un même objet tridimensionnel.

Les propriétés liées à la matrice de dispersion assurent quant à elles l'unicité de la caractérisation et sa résistance au bruit ou erreurs de discrétisation. Pour ce faire c'est l'ensemble total des points qui est pris en compte et non pas seulement les contours.

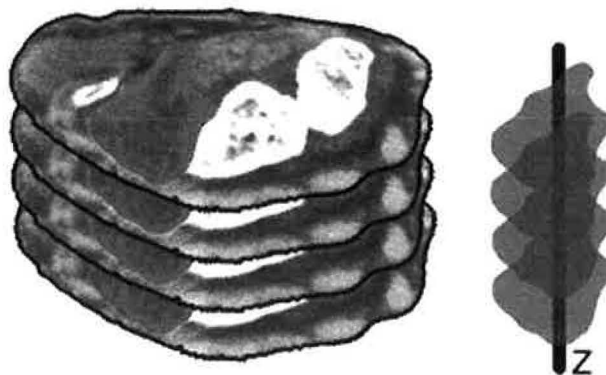


Figure 9

Pour chaque région identifiée dans un examen (ici un examen scanner des os du carpe), l'intérieur est pris en compte.

L'emploi de la distance euclidienne combinée au mode de découpage décrit en 4.6.2 – proche de la solution finale – garantit une indépendance vis-à-vis des conditions initiales.

#### **4.8. Complexité de l'algorithme progressif.**

La complexité de l'algorithme progressif est directement liée à celle des nuées dynamiques. Si  $m$  est le nombre total de points et  $k$  le niveau sémantique courant, alors

atteindre le niveau  $k+1$  requiert  $O(km)$  opérations. En effet, la meilleure minimisation du critère de scission s'obtient au prix d'essais de scission sur les  $k-1$  classes existantes, eux-mêmes suivis d'un rééquilibrage par nuées dynamiques.

Il vient donc que pour produire un squelette ellipsoïdal de profondeur  $K$  maximale, il faudra  $O(m+2m+3m+\dots+Km)$  opérations ou en d'autres termes  $O((1+2+3+4+\dots+K)m)$  opérations, ce qui se réduit en :

$$O\left(\frac{K(K-1)}{2}m\right) \approx O(K^2m)$$

où  $K$  est une constante, habituellement de l'ordre de 10.

Chaque mesure de distance entre un point et un centre de gravité nécessite l'évaluation d'une fonction qui peut s'avérer lourde (cas par exemple de la distance de Mahalanobis).

Cette technique est donc un point crucial sur lequel des voies différentes pourraient être explorées, comme le partitionnement par réseaux de neurones [18][26] ou la mise en place d'heuristiques liées au voisinage (accélération des calculs de distance aux centres de classes) [11][19].

## 6. RÉSULTATS.

Nous avons mené différentes expérimentations pour vérifier la validité et la stabilité du processus de décomposition, c'est-à-dire la qualité de caractérisation des solides tridimensionnels.

### 6.1. Mesures de stabilité.

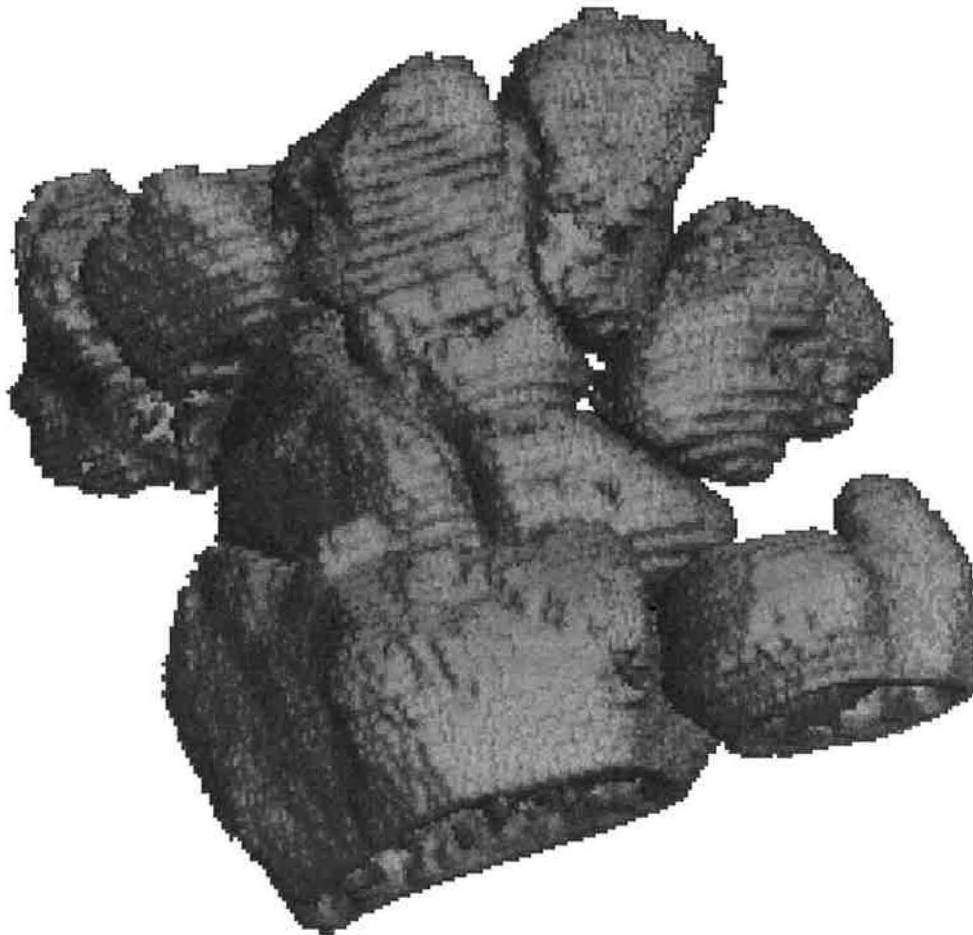
Des tests de stabilité de représentation ont été conduits sur des jeux de données issus de coupes tomodensitométriques scanner [3]. Ils mettent en évidence les propriétés d'invariance de la matrice de dispersion grâce à la prise en compte des points intérieurs à la forme. Trois catégories de problèmes ont été traités :

- *la sensibilité aux bruit et erreurs de discrétisation.* Cette catégorie regroupe les phénomènes d'anisotropie selon un axe particulier (cas des acquisitions scanner) – et plus généralement les différences d'acquisition dues à la résolution spatiale du numériseur – ainsi que les erreurs de segmentation de l'expert d'une acquisition à l'autre ;
- *l'invariance de position.* Le terme *position* désigne toute translation et rotation auxquelles peut être soumis le nuage de points initial. Nous remarquerons que la rotation peut avoir une incidence sur la numérisation selon l'anisotropie présente durant l'acquisition ;
- *la résistance au sous-échantillonnage.* Ce dernier point permet de mesurer l'efficacité de l'algorithme en fonction du facteur accélérateur demandé, c'est-à-dire de la réduction du volume des données en entrée.

#### 6.1.1. Présentation des jeux de données.

Les données suivantes ont été employées pour mener les tests de robustesse :

- *Os du carpe* : cinq examens scanner du même patient sous des angles différents (rotation et translation du poignet). Chaque examen compte les huit os du carpe à l'âge adulte, segmentés à chaque fois par un expert, le Docteur François Canovas, chirurgien orthopédique au Centre Hospitalier Universitaire Lapeyronie à Montpellier.
- *Fémurs* : examen des fémurs gauche et droit d'un même patient, segmenté par seuillage de densité dans *Corpus 2000*.



Carpe complet : on distingue l'*pulna* (cubitus) et le *radius* dans la partie inférieure de l'image. Cette représentation par *voxels* est fournie par le logiciel *Corpus 2000* (CIRAD AMAP/MEB). Les données sont issues d'un examen scanner : on peut distinguer les « strates » rendant compte de l'épaisseur des coupes tomodensitométriques. La partie supérieure est le carpe proprement dit et compte huit os (l'un d'entre eux est occulté ici) : l'*Hamatum*, le *Capitatum*, le *Triquetrum*, le *Lunatum*, le *Scaphoïde*, le *Pisiforme*, le *Trapèze* et le *Trapézoïde*. Le lecteur pourra consulter la section Bibliographie pour de plus amples renseignements concernant l'anatomie [24][25].

Les protocoles d'acquisition et de segmentations de ces examens prennent donc en compte l'erreur de numérisation liée au bruit, l'anisotropie en  $z$  ainsi que l'incertitude dans les zones périphériques des tissus à identifier.

Examen	Pixels en X	Pixels en Y	Taille des pixels	Hauteur (Z)	#coupes
Carpe	320	170	0.31mm	1mm	63
Fémurs	512	512	1mm	7.31	127

### 6.1.2. Présentation des indicateurs de variation.

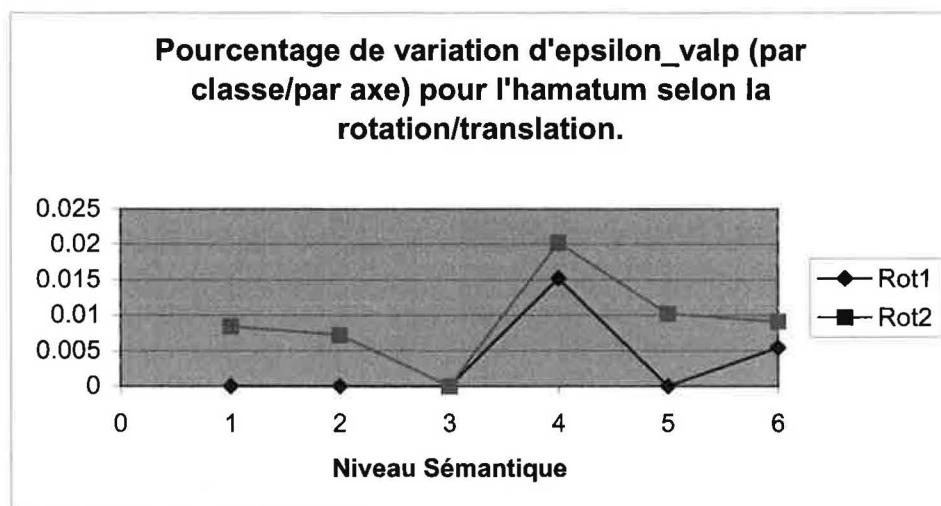
Pour observer les variations, nous mettons en place les indicateurs suivants :

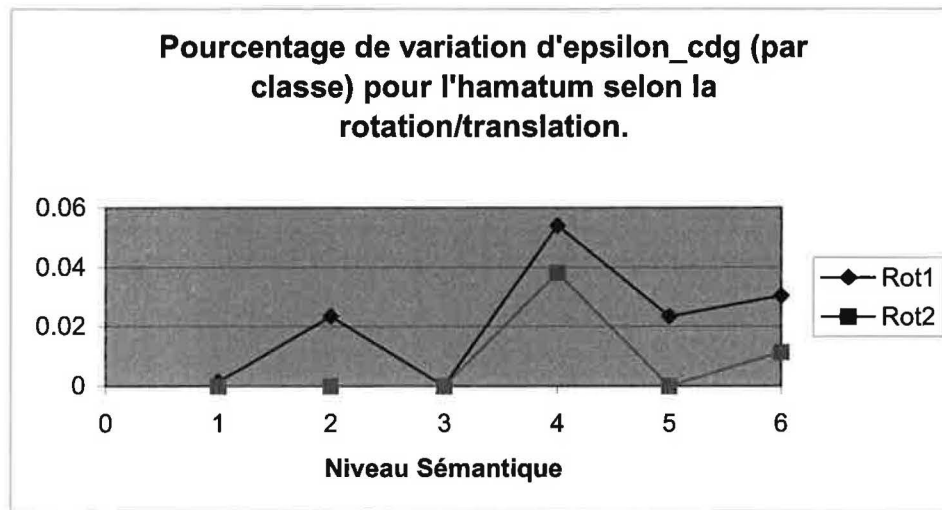
- variation de distance entre les centres de gravité des classes et le centre de gravité local de chaque objet. L'indicateur est noté  $\varepsilon_{cdg}$  (**epsilon\_cdg**). Sa variation est calculée en moyenne par classe, et convertie en pourcentage par rapport à l'amplitude moyenne observée dans l'examen de référence ;
- variations combinées des trois valeurs propres normalisées de chaque classe par rapport à un partitionnement de référence. Cet indicateur sera noté  $\varepsilon_{valp}$  (**epsilon\_valp**). Les valeurs propres sont normalisées de façon à couvrir la masse du nuage de points dans les directions des vecteurs propres. Une moyenne est ensuite réalisée par axe et par classe, puis convertie en pourcentage par rapport à l'amplitude moyenne des axes observée dans l'examen de référence.

Le nombre de classes maximal est fixé à 7 dans notre batterie de tests car une telle décomposition est largement suffisante pour caractériser un objet. Nous reviendrons sur ce point dans le chapitre *Reconnaissance de Formes par Squelette Ellipsoïdal*.

### 6.1.3. Résistance à la perturbation en position/orientation.

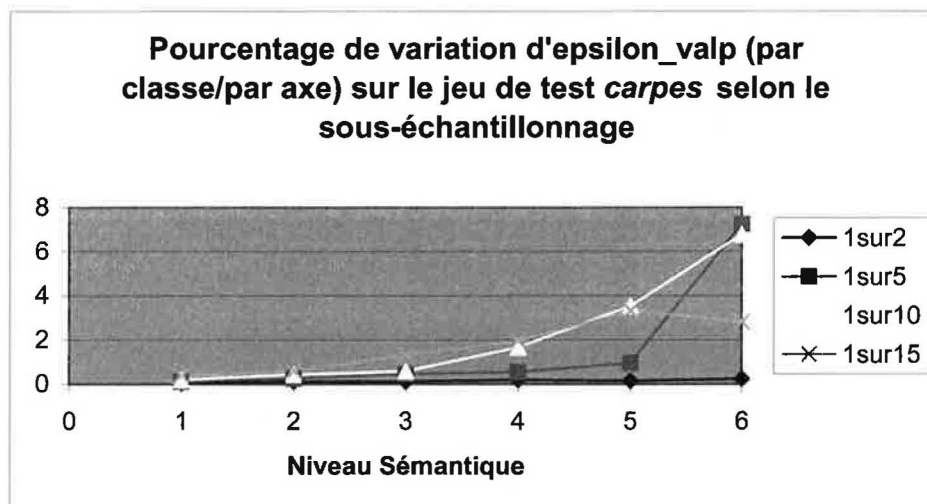
Nous avons conduit ces expérimentations en rendant le nuage de points orthotrope (isotrope dans les directions principales  $x$ ,  $y$  et  $z$ ) et en le soumettant à des rotations et translations quelconques. Dans ce cas de figure, aucune variation n'est présente au niveau des indicateurs choisis pour les tests, comme nous le montre le graphique suivant :

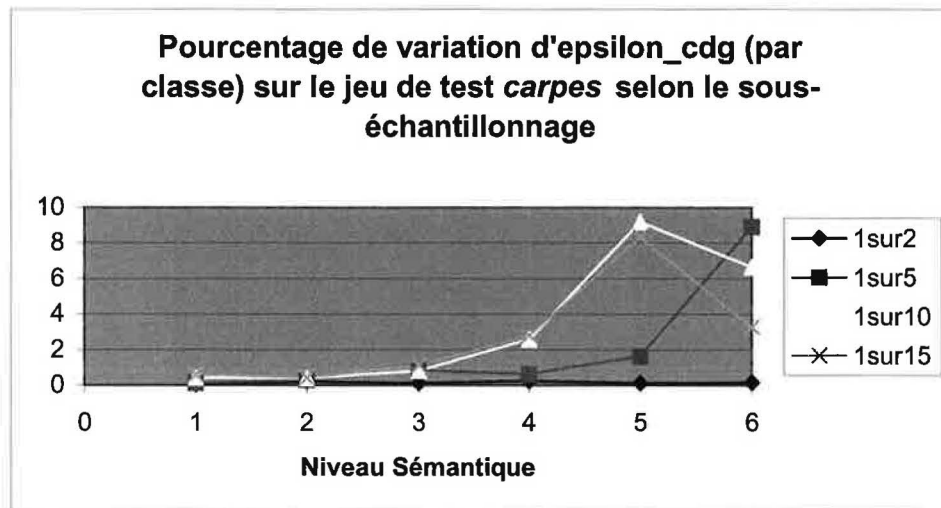




La variation des indicateurs reste inférieure à 0.05% et 0.06%, ce qui est imputable à la précision de calcul lors de la recherche des valeurs propres de la matrice de dispersion.

#### 6.1.4. Résistance au Sous-échantillonnage.

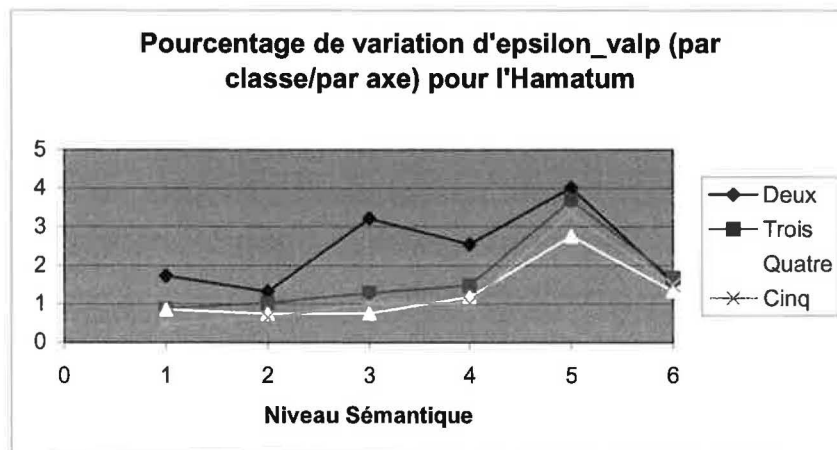


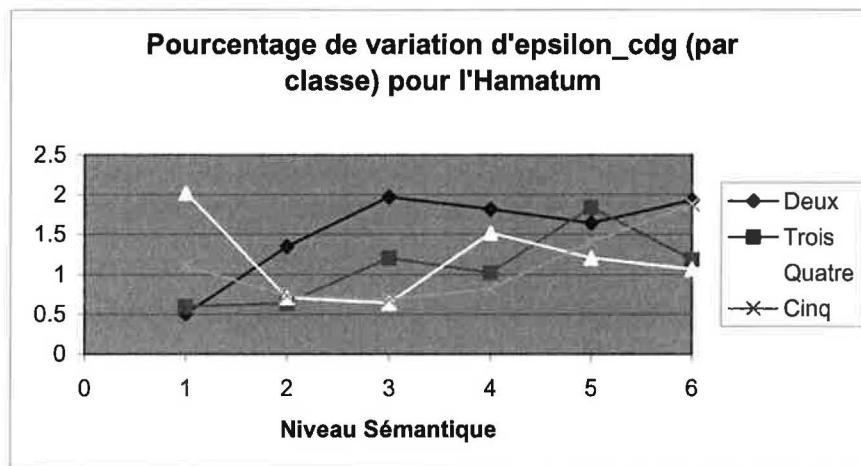


La côte de confiance dépend bien sûr du nombre de points-échantillons de chaque entité partitionnée. Si ceux-ci sont assez nombreux, les données extraites de la matrice de dispersion du nuage n'en seront que plus robustes. Des cas dégénérés peuvent survenir, même s'ils sont détectables par contrôle sur l'amplitude des valeurs propres (rang de la matrice de dispersion).

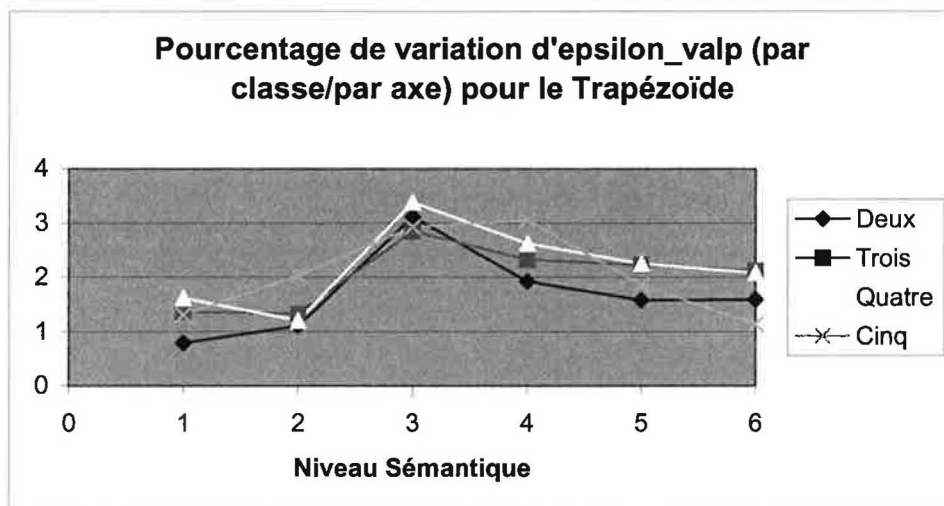
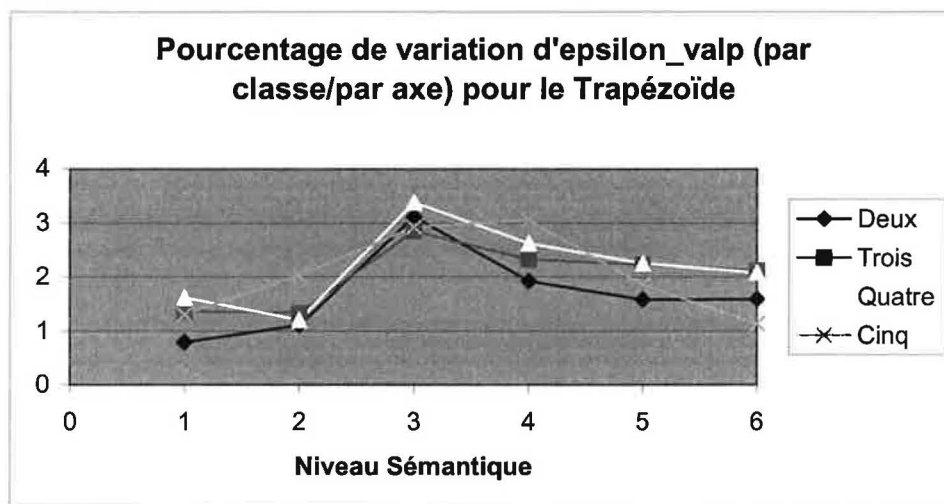
#### 6.1.5. Résistance aux erreurs de numérisation/segmentation.

Voici par exemple le cas de l'*Hamatum*, os du carpe :



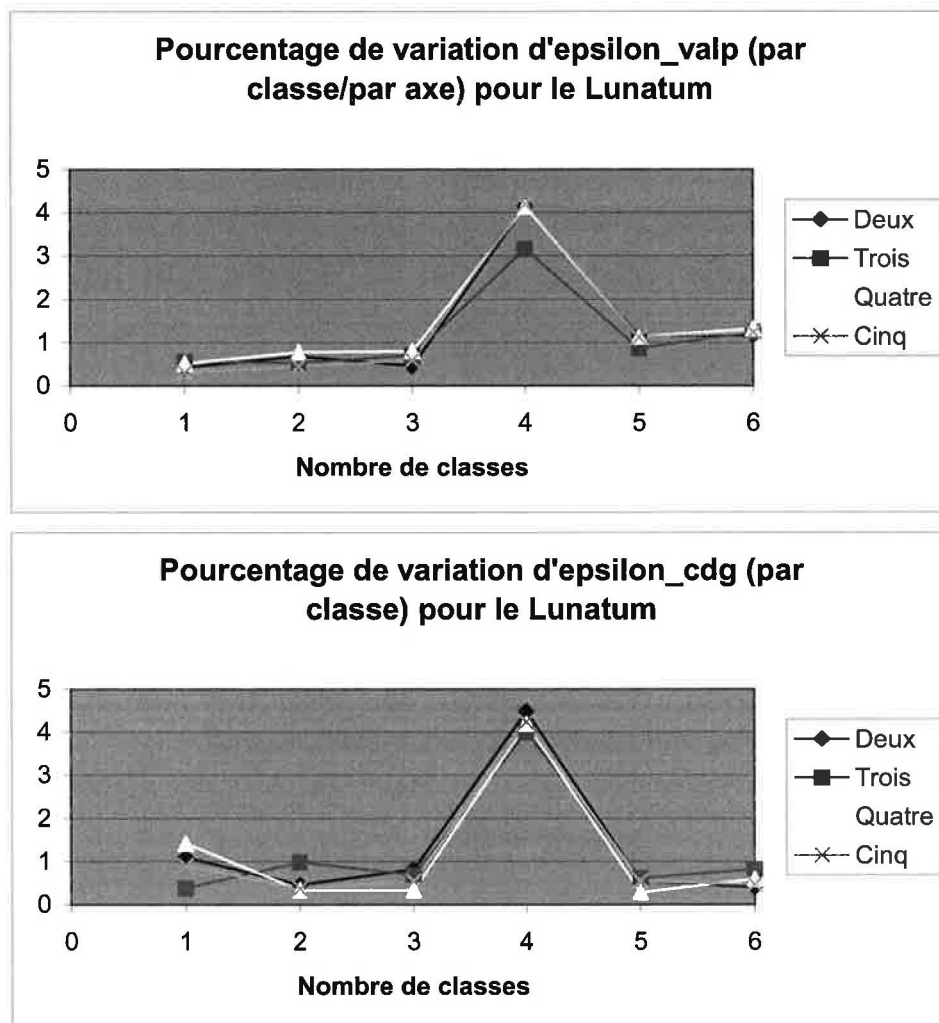


La variation observée reste inférieure à 4% sur *Hamatum*. Elle reste globalement constante, et n'augmente pas forcément de manière monotone :

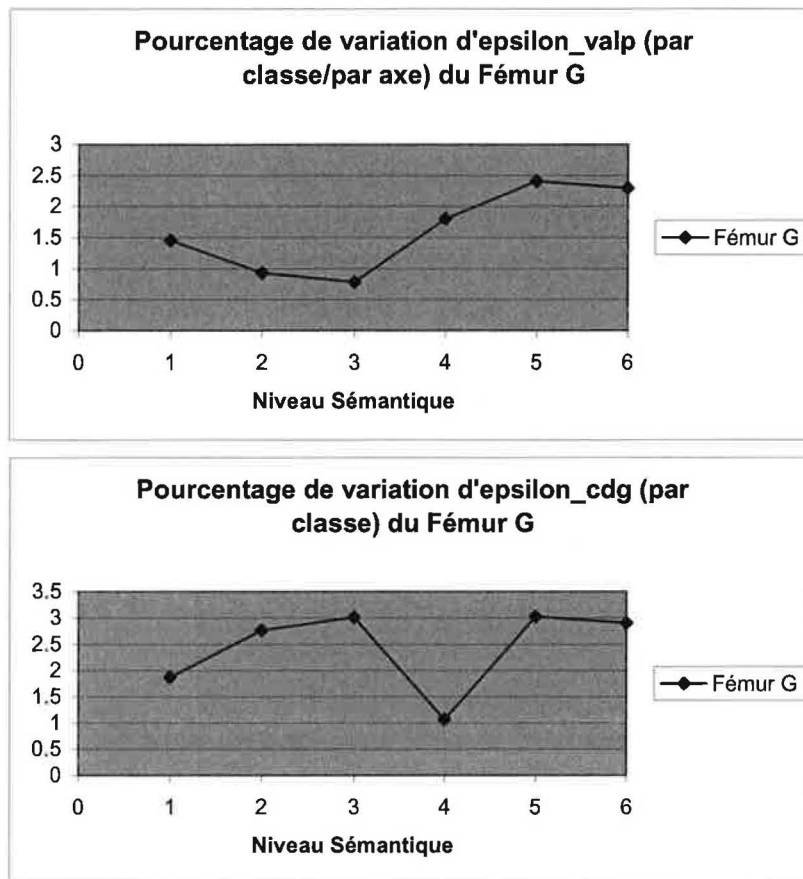




On observe dans le cas du Trapézoïde un pic de variation lors du partitionnement en quatre classes, régularisé dans les subdivisions suivantes grâce au processus d'équilibrage des nuées dynamiques. Ce phénomène peut survenir lorsque l'anisotropie verticale est trop grande et le nombre d'échantillons trop faible, ou lorsque la segmentation n'est pas rigoureusement identique, ce qui est le cas ici. Nous rappelons que pour l'examen du carpe, le facteur en  $\varepsilon$  est de l'ordre de 3. Voici une autre illustration pour le *Lunatum* :



Pour l'examen des fémurs, nous avons ôté toute prise en compte de l'orientation (pas de détection de la symétrie). Les résultats suivants sont alors obtenus :



Sur les 40 os du poignet et les fémurs, la moyenne des variations pour  $\epsilon_{valp}$  et  $\epsilon_{CdG}$  est la suivante selon le nombre de classes et relativement à l'examen de référence :

Nombre de classes	Moyenne de variation d' $\epsilon_{CdG}$ (%)	Pic (%)
2 classes	1%	3.2%
3 classes	0.8%	2%
4 classes	1.8%	4.4%
5 classes	2.1%	4.5%
6 classes	1.7%	4.3%
7 classes	1.8%	4.1%
Nombre de classes	Moyenne de variation d' $\epsilon_{valp}$ (%)	Pic (%)
2 classes	0.5%	1.6%
3 classes	0.7%	2%
4 classes	0.9%	3.4%
5 classes	1.5%	4.2%
6 classes	1.3%	4.1%
7 classes	1.5%	3.7%

Récapitulatif de la variation du partitionnement en fonction du couplage erreur de numérisation/erreur de segmentation/changement de positionnement.

## 6.2. Evolution de la variance intraclasse totale : un critère d'optimalité ?

Comme expliqué précédemment, un bon partitionnement au sens d'un critère d'optimisation doit précisément minimiser ce critère. Nous avons décidé d'observer l'évolution de la variance intraclasse totale, qui nous renseigne sur la dispersion des points autour de l'axe principal d'inertie de la classe. Minimiser cette variance revient à maximiser la variance interclasse, ce qui conduit à des classes bien différenciées tout en étant les plus homogènes possibles au sens de la dispersion et de la fonction de distance.

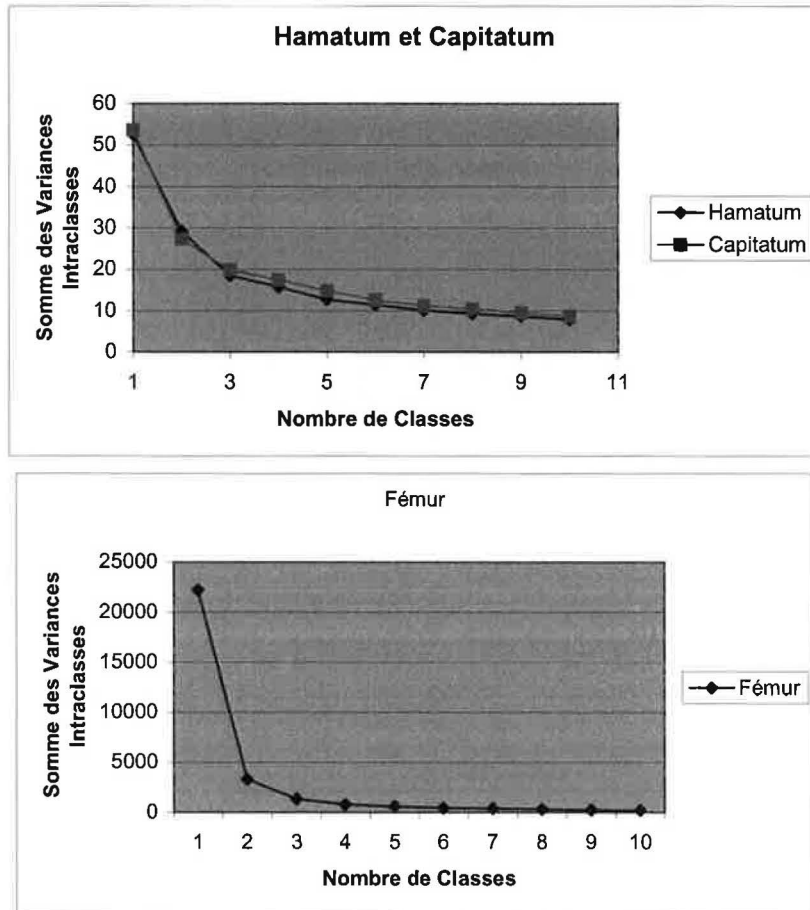


Tableau 1

Les courbes du Tableau 1 mettent en évidence l'évolution de la variance intraclasse totale sur les jeux de test présentés dans la section 6.1.1. Un seuil peut être éventuellement fixé sur la pente de la courbe afin de sélectionner un degré de qualité d'approximation géométrique adapté. La variance intraclasse totale constitue un bon indicateur dans ce contexte, à condition de considérer la dérivée de sa courbe d'évolution aux points-clés donnés et non pas sa valeur brute qui est dépendante du nombre de points par exemple. De cette manière, l'indicateur présenté ici fournit une information indépendante de la fréquence d'échantillonnage.

Il existe de nombreuses méthodes qui décrivent des indicateurs censés « juger » de la qualité du partitionnement [20][21][22]. Ils sont complexes à mettre en œuvre et souvent

très spécifiques au problème posé. Il serait néanmoins intéressant de développer ce thème plus avant, en employant des paramètres statistiques ou géométriques qui orienteraient le processus progressif de partitionnement. Ce critère d'optimalité est bien sûr étroitement lié à ce que nous nommons « critère de scission », et même si dans notre cas il est différent de celui-ci, rien n'empêche qu'il le remplace.

### **6.3. Temps de Calcul.**

Examen	Nombre de points	Profondeur Maximale	Temps
Capitatum	46228	10	2min30s
Hamatum	38767	10	2min
Fémur	152727	10	6min

Machine de référence : processeur Intel Pentium II à 233Mhz, 64Mo de RAM, système Microsoft Windows 95. Ces résultats s'entendent sans aucun sous-échantillonnage du nuage de points.

## **7. CONCLUSIONS.**

Le squelette ellipsoïdal soulève de nombreuses pistes d'investigation. Le lecteur sera conscient par exemple de l'absence de définition intrinsèque de la gauche et de la droite – à moins de définir explicitement une direction - : en effet, le squelette ellipsoïdal d'un fémur droit ou gauche sera exactement le même. Pour répondre à ces ouvertures, nous avons choisi de nous orienter vers un système générique, et ce même dans l'implantation que nous proposons. En effet, ce sont toujours des paramètres simples et intuitifs qui conduisent le découpage des objets tridimensionnels. La structure du squelette ellipsoïdal autorise néanmoins de multiples possibilités d'amélioration et de spécialisation en fonction des contextes de recherche. Les buts avoués restent toujours la représentation synthétique, hiérarchique ainsi que la caractérisation des entités volumiques, quelles que soient leurs origines. Nous avons pour cela tenté d'apporter une pierre à l'édifice en développant un cadre formel prêt à évoluer. Les chapitres suivants développeront les utilisations du squelette ellipsoïdal, et démontreront son utilité à travers des exemples de mise en place d'algorithmes particuliers, adaptés à différents domaines.

## 8. BIBLIOGRAPHIE.

- [1] Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Partitionnement Hiérarchique d'un Nuage de Points par Squelette Ellipsoïdal* », 1998, 6ièmes Journées de l'Association Française d'Informatique Graphique, Dunkerque, France.
- [2] Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Application of Ellipsoidal Skeleton to Organ Geometric Modeling from CT-scan Images.* », 1999, SPIE Conference on Medical Imaging '99, San Diego, California.
- [3] Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Hierarchical Pattern Analysis and Recognition Using Ellipsoidal Skeleton.* », 2000, Fourth International Conference in Computer Graphics and Artificial Intelligence, 3IA'2000, Limoges, France.
- [4] Murray R. Spiegel, « *Théorie et Applications de la Mécanique Générale.* », Série Schaum, Mac Graw Hill, 1972.
- [5] D. Forsyth, J. L. Mundy, A. Zisserman, C. Coelho, A. Heller and C. Rothwell, « *Invariant Descriptors for 3D Object Recognition and Pose.* », IEEE Transactions on PAMI, 13 :10, pp. 971-991, October, 1991.
- [6] Y. Guédon, « *Variables Aléatoires Discrètes, Processus Stochastiques à Temps Discret et à Espace d'Etats Discret.* », Rapport Interne, CIRAD, Programme Modélisation des Plantes, 1999.
- [7] A. Leonardis, A. Jaklic and F. Solina, « *Superquadrics for Segmenting and Modeling Range Data.* », IEEE Transactions on PAMI, 19 :11, pp. 1289-1295, November, 1997.
- [8] P. Bajcsy, N. Ahuja, « *Location- and Density-Based Hierarchical Clustering Using Similarity Analysis.* », IEEE Transactions on PAMI, 20 :9, pp. 1011-1015, Short Papers, September, 1998.
- [9] I. Pitas, A. N. Venetsanopoulos, « *Morphological Shape Decomposition.* », IEEE Transactions on PAMI, 12 :1, pp. 38-45, January, 1990.
- [10] P. Borianne, « *Cepha 3Dt-version 2.0.* », Chapitre « Outils Mathématiques. », CIRAD-MEB, 1998.
- [11] M. Murphy, S. Skiena, « *A Study of Data Structures for Orthogonal Range and Nearest Neighbor Queries in High Dimensional Spaces.* », CSE 523/524 Master's Project, Department of Computer Science, State University of New York at Stony Brook, NY 11790-4400.
- [12] « *Recherche Opérationnelle/Operations Research.* », 0399-0559/98/04, Elsevier, Paris, 1998.
- [13] E. Diday, « *Optimisation en Classification Automatique et Reconnaissance des Formes.* », RAIRO, Novembre, 1972.
- [14] M. Jambu, « *Classification Automatique pour l'Analyse des Données.* », Dunod, 1978.
- [15] S. Regnier, « *Sur Quelques Aspects Mathématiques des Problèmes de Classification.* », ICC Bull., vol. 4, pp. 175-191, 1965.
- [16] K. S. Younis, « *Weighted Mahalanobis Distance for Hyper-Ellipsoidal Clustering.* », PhD. Thesis, Department of the Air Force Institute of Technology, RJAf, AFIT/GE/ENG/96D-22, Wright-Patterson Air Force, December, 1996.

- [17] D. R. Wilson, T. R. Martinez, « *Improved Heterogeneous Distance Functions.* », Journal of Artificial Intelligence Research, vol. 6, pp. 1-34, AI Access Foundation and Morgan Kaufmann Publishers, 1997.
- [18] B. D. Ripley, « *Neural Networks and Related Methods for Classification.* », Journal of The Royal Statistical Society, Series B, vol. 56, pp. 409-456, 1994.
- [19] S. Byers, A. E. Raftery, « *Nearest Neighbor Clutter Removal for Estimating Features in Spatial Point Processes.* », Journal of the American Statistical Association, 1998.
- [20] R. E. Kass, A. E. Raftery, « *Bayes Factor.* », Journal of the American Statistical Association, vol. 90, pp. 773-795, 1995.
- [21] A. J. Scott, M. J. Symons, « *Clustering Methods based on Likelihood Ratio Criteria.* », Biometrics, vol. 27, pp. 387-397, 1971.
- [22] L. Kaufman, P. J. Rousseuw, « *Finding Groups in Data.* », Wiley, 1990.
- [23] J. A. Hartigan, « *Clustering Algorithms.* », Wiley, 1975.
- [24] P. Kamina, J-P. Francke, « *Anatomie : Introduction à la Clinique. Arthrologie des Membres.* », Maloine S.A. Editeurs, 1988.
- [25] J-P. Chevrel, J-P. Guéraud and J-B. Lévy, « *Anatomie Générale.* », 4<sup>ème</sup> Edition, Masson, 1986.
- [26] N. Ueda, « *Optimal Linear Combination of Neural Networks for Improving Classification Performance.* », IEEE Transactions on PAMI, 22 :2, pp. 207-215, February, 2000.

## **6. Reconstruction Géométrique par E- Squelette.**



## **1. INTRODUCTION.**

La complexité sans cesse croissante des objets à manipuler découle essentiellement des progrès réalisés dans les modalités d'acquisition. Les numérisations par laser, scanner, IRM ou palpeurs par exemple bénéficient d'une plus grande précision alliée à une fréquence d'échantillonnage sans cesse plus grande.

De ce fait le flot d'informations géométriques à gérer est alourdi, et même si les calculateurs augmentent en puissance de traitement, il reste évident que des heuristiques doivent être mises en place pour réduire le volume des données. Cependant, la compression de données va à l'encontre de la fidélité pour peu que l'on choisisse une technique destructrice pour obtenir de meilleurs résultats – d'où un compromis difficile à trouver entre la compression et la fidélité.

La visualisation scientifique s'inscrit totalement dans cette problématique, puisqu'à chaque instant il faut pouvoir affiner l'information tout en se donnant la possibilité de l'observer dans son ensemble. L'interactivité dans la manipulation est primordiale : elle doit faciliter la compréhension et permettre à l'expert de se consacrer uniquement à son travail.

Pour ce faire, nous proposons avec le E-Squelette une structuration de l'information, comme présenté dans le chapitre précédent, à laquelle vient s'ajouter un affichage dynamique. Nous montrerons d'abord dans ce chapitre comment la structure hiérarchique et analytique du E-Squelette permet une régulation constante du flux d'information. Ensuite les thèmes de compression des données géométriques et de transmission par réseau seront discutés. Enfin, nous montrerons les capacités de reconstruction surfacique fine du E-Squelette.

## 2. MODÈLE INTERNE GÉOMÉTRIQUE.

Nous présentons ici le modèle mathématique sous-jacent supportant la représentation visuelle d'un E-Squelette. Il sera étudié de manière théorique puis nous proposerons notre structure de données.

### 2.1. Les primitives.

Une fois le nuage de points partitionné en classes, l'idée de choisir un objet géométrique simple pour représenter - au moins approximativement - chacune d'elle semble naturelle [17][19][23]. Chacun de ces objets de base est appelé *primitive*, puisqu'il décrit une forme de la manière la plus réduite possible, sans possibilité d'affinage.

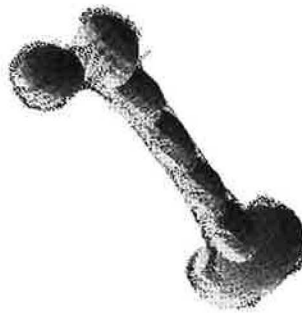


Figure 1

Les ellipsoïdes montrés dans la Figure 1 sont des primitives adéquates. Il suffit de les positionner sur les centres de gravité de chaque classe et de les aligner sur les trois axes principaux d'inertie avec les paramètres de mise à l'échelle qui conviennent :

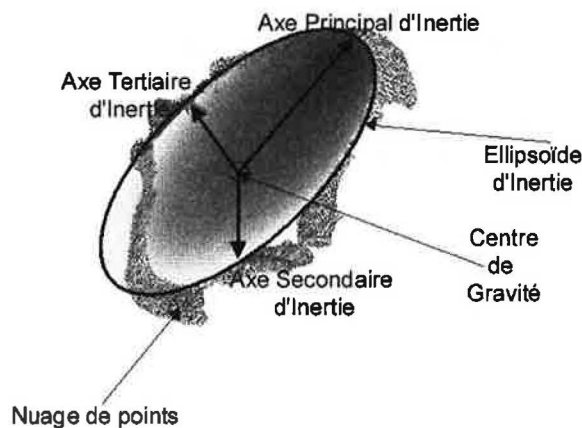


Figure 2

La primitive montrée en Figure 2 a un sens mécanique puisqu'elle représente l'*ellipsoïde d'inertie*. Notons que chaque rayon  $r_i = 2\sqrt{\lambda_i}$ , ceci en regard de la loi des grands nombres (au moins 95% des points sont couverts par l'ellipsoïde d'inertie).

L'ellipsoïde se définit par l'équation classique suivante :

$$d_i(x, y, z) \stackrel{def}{=} \frac{x^2}{a_i^2} + \frac{y^2}{b_i^2} + \frac{z^2}{c_i^2} - 1 = 0$$

Équation 1

où  $a_i$ ,  $b_i$  et  $c_i$  sont les amplitudes des trois vecteurs principaux d'inertie (dans l'ordre décroissant) de la  $i^{\text{ème}}$  primitive. L'Équation 1 constitue en réalité un cas particulier de la famille des *superquadriques*, définies par l'équation suivante :

$$f_i(x, y, z) \stackrel{def}{=} \left( \left( \frac{x}{a_i} \right)^{\frac{2}{\varepsilon_1}} + \left( \frac{y}{b_i} \right)^{\frac{2}{\varepsilon_1}} \right)^{\frac{\varepsilon_1}{\varepsilon_2}} + \left( \frac{z}{c_i} \right)^{\frac{2}{\varepsilon_2}} - 1 = 0$$

Équation 2

avec  $\varepsilon_1$  et  $\varepsilon_2$  les coefficients respectifs dits *est/ouest* et *nord/sud*. Ces derniers agissent sur la forme de l'ellipsoïde de la façon suivante :

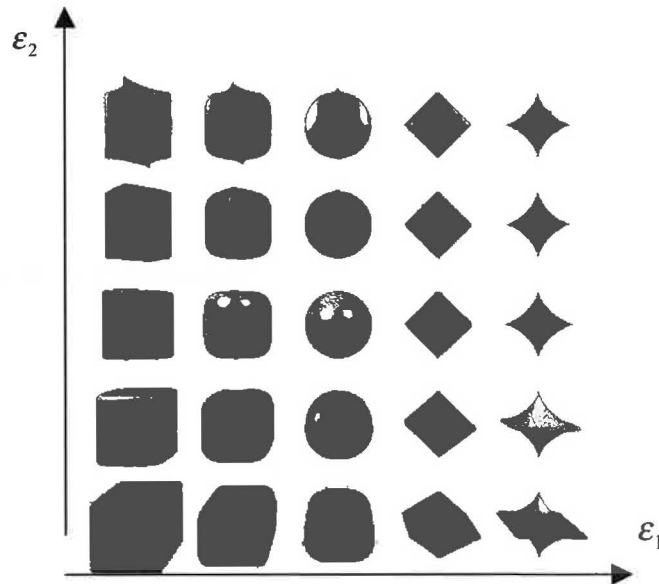


Figure 3

Les superquadriques offrent un vaste choix de primitives géométriques : ceci permettra un meilleur contrôle surfacique lors de la reconstruction géométrique ainsi qu'une caractérisation continue de la forme des classes (donc injectable dans la distance entre objets présentée au chapitre 7).

Nous expliquerons plus avant dans ce chapitre comment choisir au mieux les différents paramètres de la superquadrique.

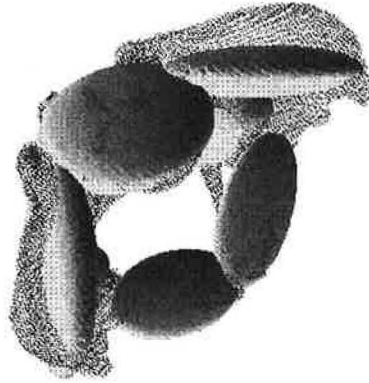


Figure 4

La Figure 4 est un E-Squelette à cinq classes de l'os iliaque. Les primitives sont des ellipsoïdes dans cet exemple.

## 2.2. Les surfaces implicites.

Nous rappelons à présent quelques éléments décrivant les surfaces implicites.

Lorsque les primitives sont positionnées, il est possible d'améliorer encore la représentation par primitives en les recouvrant d'une « peau » comme indiqué dans la figure suivante :

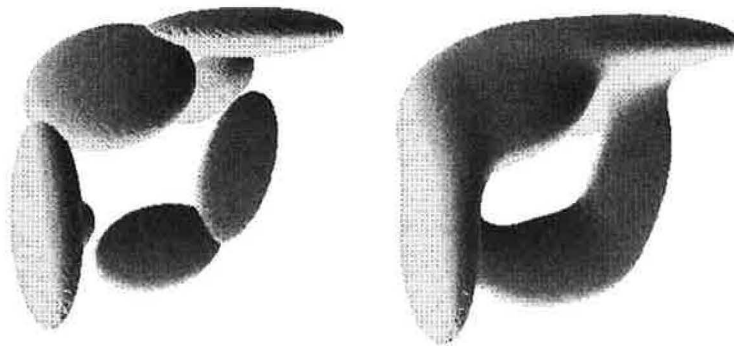


Figure 5

Pour ce faire, nous avons choisi le modèle des surfaces implicites qui permet de raccorder simplement deux surfaces de manière continue, sans employer de techniques complexes comme c'est le cas pour les surfaces paramétrées.

Une surface implicite est définie par l'équation :

$$f_i(x,y,z)=c$$

Équation 3

où  $c$  est une constante réelle nommée *isovaleur*. Cette surface est le contour, ou la frontière, d'un objet 3D dont l'intérieur est l'ensemble des points  $(x, y, z)$  tels que  $f_i(x, y, z) < c$  et l'extérieur l'ensemble des points vérifiant  $f_i(x, y, z) > c$ . Par abus de langage, on confond la surface implicite (la frontière) et l'intérieur de l'objet ainsi défini. Plusieurs modèles de surfaces implicites existent [21], chacun avec ses avantages et ses inconvénients particuliers. La fonction de l'Équation 3 est appelée *fonction de distance*.

La surface est alors générée en cherchant dans l'espace  $R^3$  les points vérifiant l'Équation 3. La Figure 6 nous montre un exemple en deux dimensions :

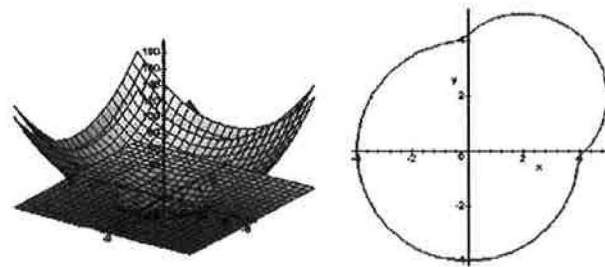


Figure 6

Courbe implicite définie par  $f(x,y)=1$ . La partie gauche montre la surface d'équation  $z=f(x,y)$  et le plan d'équation  $z=1$ . Leur intersection est la courbe sur la partie droite de la figure.

Ainsi, pour produire par exemple la surface générée par l'Équation 2, nous poserons  $c=0$ , ce qui donne :

$$f_i(x, y, z) = 0$$

Équation 4

Un avantage important d'une telle relation est la détermination de l'appartenance d'un point à une surface, à son intérieur ou à son extérieur dans le cas d'une surface fermée. En effet, soit un point  $P(x_P, y_P, z_P)$  de  $R^3$ , les relations suivantes indiquent sa position relativement à  $f_i$  :

$f_i(x_P, y_P, z_P) > 0$  indique que  $P$  est à l'extérieur de la surface

$f_i(x_P, y_P, z_P) = 0$  indique que  $P$  est sur la surface

$f_i(x_P, y_P, z_P) < 0$  indique que  $P$  est à l'intérieur de la surface

Équation 5

Remarque :

Un tel test est beaucoup plus difficile à réaliser lorsque la surface de l'objet est décrite par un assemblage de carreaux (*patches*) de surfaces paramétrées. De fait, tout ensemble de surfaces paramétriques ne définit pas forcément la frontière d'un objet cohérent.

Pour combiner géométriquement les primitives, il existe des fonctions dites *booléennes* qui agissent sur les fonctions  $f_i$ . Soient  $f_1$  et  $f_2$  les fonctions définissant deux primitives implicites dont la surface est définie selon la même isovaleur  $c$ . Les fonctions booléennes sont alors définies comme suit [5] :

$$\begin{aligned} \text{union}_{\text{booléenne}}(f_1, f_2) &\stackrel{\text{def}}{=} \min(f_1(x, y, z), f_2(x, y, z)) = c \\ \text{intersection}_{\text{booléenne}}(f_1, f_2) &\stackrel{\text{def}}{=} \max(f_1(x, y, z), f_2(x, y, z)) = c \end{aligned}$$

Équation 6

Les fonctions booléennes d'union et d'intersection.

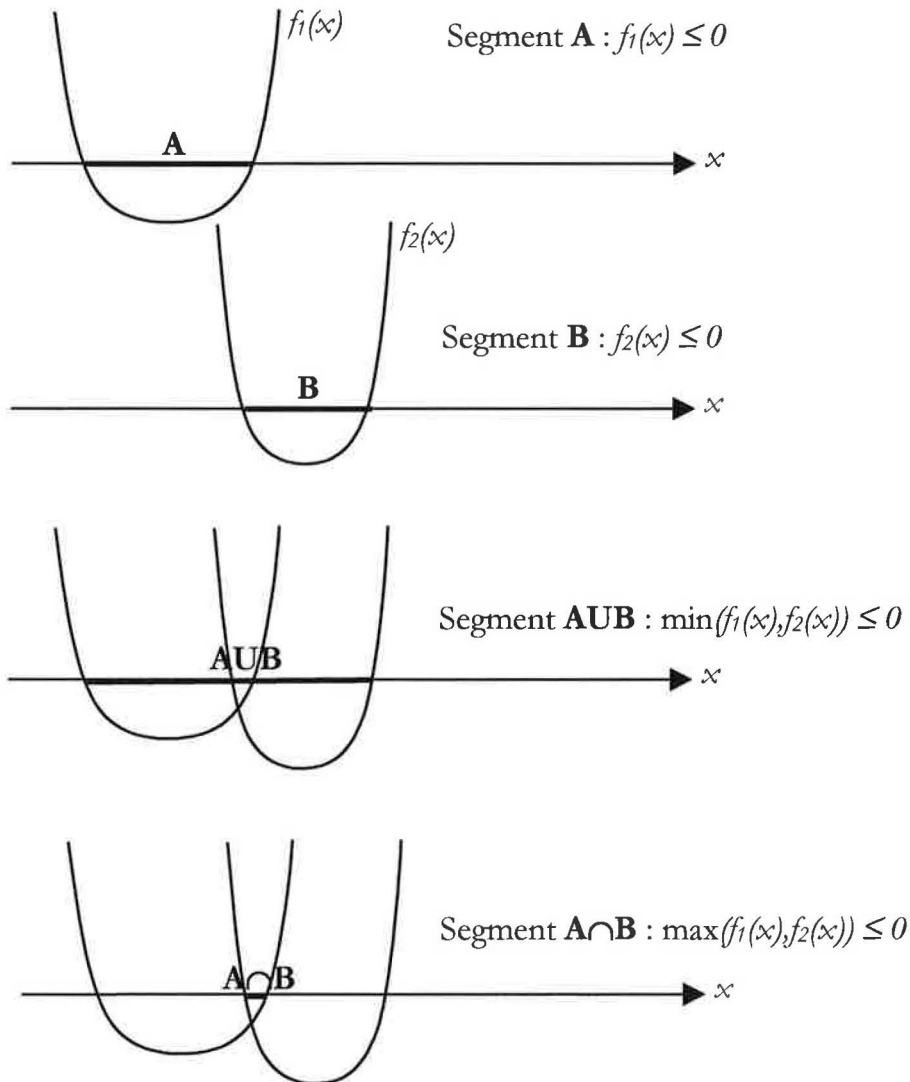


Figure 7

Explication de l'union booléenne.

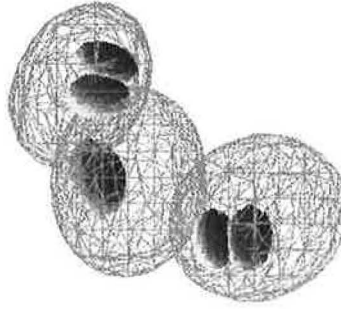


Figure 8

Union booléenne exacte entre primitives implicites. C'est bien un objet unique qui est représenté ici.

Les fonctions présentées dans l'Équation 6 ne sont bien sûr pas dérivables en tout point puisqu'elles peuvent présenter des courbes d'inflexion dans les zones d'intersection des deux surfaces. La Figure 8 montre la reconstruction d'un foie tumoral (les lésions sont en bleu) avec union exacte entre les primitives.

Les fonctions booléennes dites *de mélange* sont définies par :

$$\begin{aligned} \text{intersection}_{\text{mélange}}(f_1, f_2) &= -\alpha_1 e^{-\beta_1 f_1} - \alpha_2 e^{-\beta_2 f_2} \\ \text{union}_{\text{mélange}}(f_1, f_2) &= \alpha_1 e^{\beta_1 f_1} + \alpha_2 e^{\beta_2 f_2} \end{aligned}$$

Équation 7

où les constantes  $\alpha_i, \beta_i$  sont réelles positives. Chaque élément  $\alpha_i e^{\beta_i f_i}$  est nommé *fonction de potentiel* [28]. La constante  $c$  est alors mise à 1.



Figure 9

Voici la même reconstruction que la Figure 8 mais en employant une union de mélange entre les primitives.

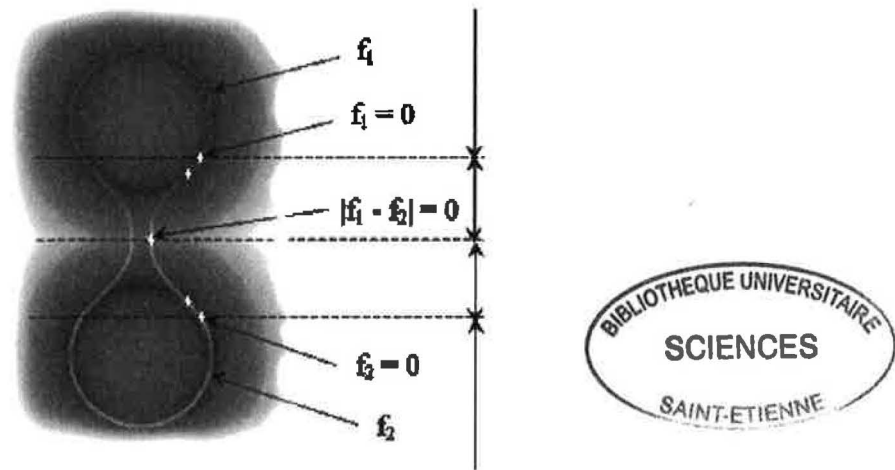


Figure 10

Le formalisme lié aux fonctions booléennes continues simule l'émission d'un potentiel par les noyaux (en rouge et jaune). Les zones où la somme des potentiels émis est égale à une constante (isovaleur) définissent une surface dans l'espace (le cas présenté ici n'emploie pas d'exponentielle).

Les fonctions continues présentées dans l'Équation 7 produisent des surfaces dérivables en tout point, ce qui constitue une propriété intéressante pour l'*ajustement surfacique* comme nous le verrons plus loin. Elles présentent toutefois deux inconvénients :

- les fonctions exponentielles ont une influence dans l'espace entier. Il faut prendre en compte ce phénomène en indiquant pour chaque objet quelles sont les primitives mises en jeu uniquement pour celui-ci (pas de prise en compte des autres objets) ;
- les fonctions exponentielles requièrent un coût calculatoire assez élevé, ce qui oblige à minimiser autant que possible le nombre de primitives ;
- enfin il y a un risque de débordement numérique à cause de l'exponentiation (*overflow*).

Il existe d'autres fonctions d'union booléenne qui sont polynomiales [5][40][42]. La plupart présentant des problèmes de dérivabilité, nous avons opté pour l'utilisation des fonctions employant l'exponentielle. Des mesures ont été prises pour éviter les problèmes cités dans le paragraphe précédent, comme nous le verrons tout au long de ce chapitre.

Pour organiser les primitives et gérer les opérations booléennes qui leur sont appliquées, nous introduisons à présent l'*arbre géométrique du E-squelette*.

### 2.3. L'Arbre Géométrique du E-Squelette.

L'arbre géométrique [33] s'inspire classiquement de la *géométrie solide constructive* [30]. Pour un niveau sémantique donné, il traduit l'expression analytique [7][22] de la forme surfacique associée de manière constructive.



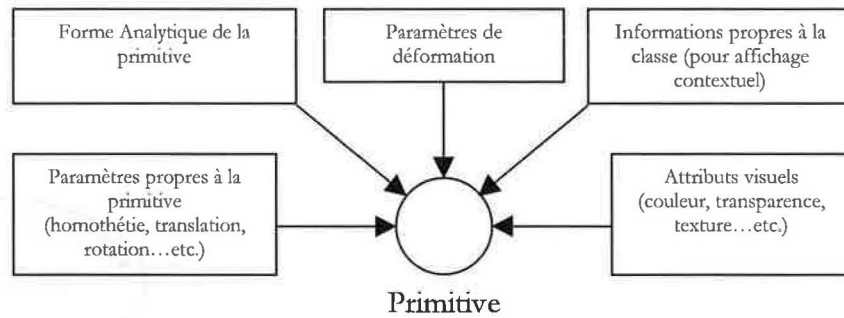


Figure 11

Le nœud d'une primitive regroupe tous les paramètres nécessaires à sa visualisation.

Il est important de remarquer que la forme analytique de la primitive regroupe non seulement la fonction de distance (comme dans l'Équation 1 ou l'Équation 2) mais aussi la fonction de potentiel. Ce choix autorise une plus grande souplesse durant la phase d'ajustement surfacique (voir section 2.4).

Les paramètres de déformation sont dépendants du modèle choisi. Nous présenterons en détail la fonction de déformation que nous avons mise en place en section 2.4. La déformation ne s'applique qu'à la primitive concernée, de manière globale sur celle-ci car nous voulons éviter toute expression paramétrique ou par éléments finis.

Le processus d'évaluation d'un nœud est alors le suivant :

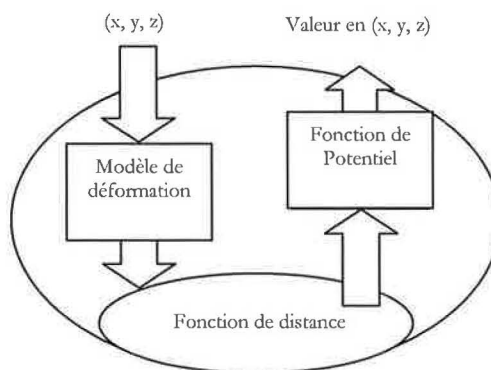


Figure 12

Lorsqu'une opération booléenne est requise entre deux primitives – sans perte de généralité, nous ne considérons que le cas binaire –, l'arbre prend la forme suivante :

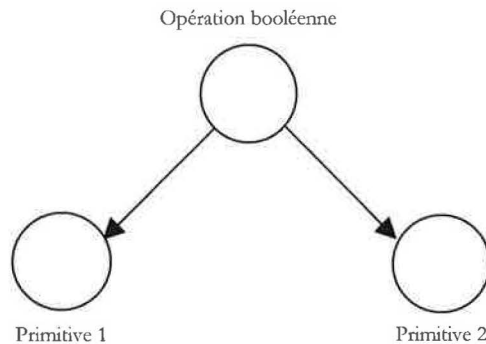


Figure 13

Le nœud d'opération d'union booléenne, par exemple, combine les valeurs de sortie des primitives (selon l'Équation 7). Son fonctionnement est donc identique à celui présenté en Figure 12, à ceci près qu'il n'y a pas de fonction de distance.

Pour évaluer en un point  $(x, y, z)$  la fonction d'union entre deux primitives, il suffit alors d'exécuter le processus d'évaluation sur le nœud de l'opération booléenne en question, les sous-évaluations étant résolues récursivement :

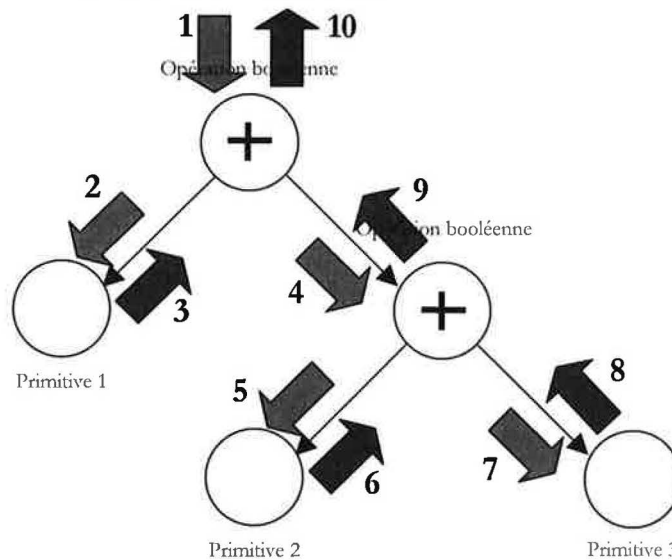


Figure 14

La Figure 14 montre l'organisation de la combinaison de trois primitives. Chaque nœud portant une opération booléenne nécessite l'évaluation de ses fils (opérandes) avant de renvoyer lui-même une valeur en  $(x, y, z)$ . Les chiffres indiquent l'ordre d'évaluation de l'arbre, les flèches rouges indiquant l'injection des coordonnées et les flèches bleues le renvoi de la valeur adéquate. Les feuilles (primitives) renvoient les valeurs de potentiels adéquates, alors que les nœuds booléens réalisent une simple somme des valeurs reçues.

## **2.4. L'Ajustement Surfamique.**

Nous allons à présent nous intéresser à l'ajustement surfamique, qui désigne la déformation de la surface de manière à approcher au mieux les points de contour du solide tridimensionnel. Cette technique peut-être décomposée en trois étapes :

- obtention d'une solution initiale « satisfaisante », qui ne soit pas trop « éloignée » du résultat ;
- usage d'un modèle de déformation adapté, avec d'éventuelles contraintes de conservation de la forme analytique ;
- utilisation d'une méthode d'ajustement robuste qui va rechercher la solution pas à pas.

Nous présenterons les différentes composantes du problème de l'ajustement surfamique, puis – une fois ces notions assimilées par le lecteur – nous décrirons notre technique complète intégrant tous ces aspects (section 2.4.8). On pourra se reporter à cette dernière section directement si l'on est familier des problèmes d'optimisation numérique.

### **2.4.1. Obtention de la Solution Initiale.**

La qualité de la solution initiale conditionne totalement la vitesse de convergence de l'algorithme d'ajustement et peut lui éviter d'être piégé dans des minima locaux.

Il nous faut d'abord trouver le niveau sémantique qui minimise l'erreur en chaque point de la surface de l'objet (voir section 2.4.5). L'obstacle majeur étant le coût de l'évaluation des fonctions de distance et de potentiel, un sous-échantillonnage des points de contour est à envisager.

Nous avons ainsi développé une méthode automatique d'obtention initiale qui évite l'évaluation de l'arbre géométrique tout en offrant un bon compromis sur la qualité de la solution initiale. Pour cela nous employons l'indicateur de *variance intraclasse*, car il nous informe sur la qualité du partitionnement : sa variation est mesurée d'un niveau sémantique  $n$  à un niveau sémantique  $n+1$ . Lorsque cette variation est faible relativement aux diminutions précédentes, nous estimons qu'un niveau adéquat a été atteint.

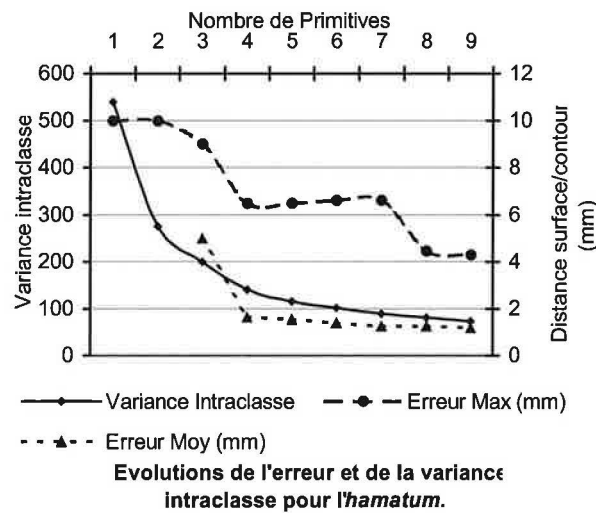


Tableau 1

Le Tableau 1 montre l'évolution de l'erreur relativement à celle de la variance intraclasse, en moyenne sur les cinq examens du carpe pour l'*hamatum*. Le premier critère de sélection du niveau sémantique est donc la faible variation de la variance intraclasse (ici à partir de 7 classes par exemple).

Il s'agit ensuite de choisir un nombre de classes raisonnable à partir du niveau sémantique trouvé :

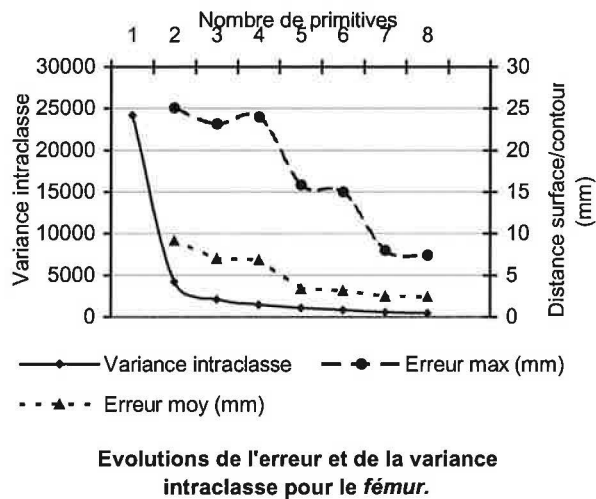


Tableau 2

L'évolution quasi-stationnaire pour 7 classes semble indiquer dans le Tableau 2 que ce niveau sémantique conviendra pour la solution initiale dans le cas du fémur.

Cette heuristique très simple s'est avérée efficace durant nos expérimentations, présentant l'avantage d'un coût calculatoire quasi-nul et d'une sélection instantanée du point de départ de l'ajustement surfacique.

#### 2.4.2. Choix de la primitive.

Comme nous l'avons vu dans la section 2.1, les superquadriques offrent une grande variété de formes. Deux choix sont possibles :

- proposer de manière fixe – c'est-à-dire pour des couples déjà définis  $(\varepsilon_1, \varepsilon_2)$  – la primitive la plus proche des points de contour (voir la section 2.4.5 pour la mesure de l'erreur de l'approximation géométrique) ;
- optimiser numériquement le couple  $(\varepsilon_1, \varepsilon_2)$ , ce qui est plus coûteux en temps de calcul mais qui présente le double avantage d'être plus précis et de pouvoir ensuite employer  $\varepsilon_1$  et  $\varepsilon_2$  comme mesure de ressemblance entre classes (voir le chapitre concernant la reconnaissance de formes par E-Squelette).

#### 2.4.3. Un modèle « local » de déformation.

Pour conserver la formulation analytique, nous avons opté pour une déformation globale de chaque primitive. De cette manière, il est possible de contrôler plus ou moins localement l'ajustement surfacique. Nous nous sommes inspirés du modèle des *vibrations modales* [13], qui contrôle les déformations au travers d'un vecteur de dimension 30 :

$$\boldsymbol{\mu}_i = (t_x \quad t_y \quad t_z \quad \varphi \quad \theta \quad \psi \quad a \quad b \quad c \quad \mu_1 \quad \dots \quad \mu_p)$$

Équation 8

Chaque coordonnée décrit une transformation selon les trois axes locaux de la primitive (son trièdre d'inertie). Les modes basses fréquences concernent la translation, la rotation et l'homothétie, les modes hautes fréquences gérant la torsion, le cisaillement...etc.

Le vecteur  $\boldsymbol{\mu}_i$  définit une matrice de déformation de dimensions  $3 \times 3$ , notée  $\mathbf{D}_i(\boldsymbol{\mu}_i, x, y, z)$ . Un point  $P$  appartenant au contour de la  $i^{\text{ème}}$  primitive déformée vérifie alors la relation suivante :

$$f_i(\mathbf{D}_i^{-1}(\boldsymbol{\mu}_i, P).P) = 0$$

Équation 9

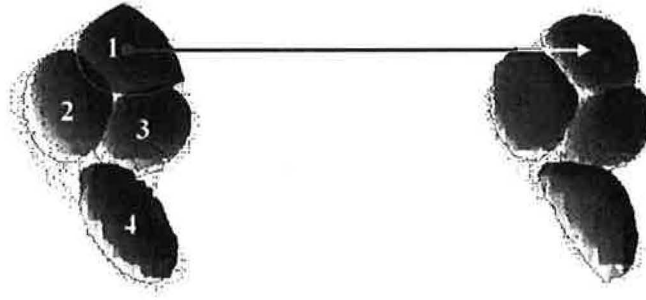


Figure 15

La Figure 15 illustre quelques possibilités de déformation du modèle présenté dans cette section. Chaque primitive (de 1 à 4 ici) subira ainsi une modification relativement à son contour local : à droite nous pouvons constater l'effet de non recouvrement (ou de faible recouvrement) entre les primitives géométriques. Le contour local est déterminé de manière triviale lorsque les points sont extraits de coupes scanner puisqu'il suffit de ne sélectionner que les contours 2D dans chaque image. Pour un nuage quelconque, ce problème est plus difficile.

D'autres modèles de déformation existent [26], et peuvent remplacer ou même s'ajouter à l'existant. Ils pourront alors être organisés en chaînes qui s'appliqueront successivement aux coordonnées d'entrée en chaque sommet.

#### 2.4.4. Fusion des primitives.

Une fois chaque primitive ajustée avec ses points de contour, les primitives sont fusionnées grâce aux fonctions d'unions booléennes continues. Il faut pour cela réguler simultanément tous les couples de valeurs  $(\alpha_i, \beta_i)$  de chaque nœud d'union. Cette opération est réalisée par une méthode d'optimisation numérique, dont nous proposons quelques exemples dans la section suivante (2.4.6).

#### 2.4.5. Evaluation de l'approximation géométrique.

La surface doit approcher au mieux les points de contour de l'objet. C'est pour cela qu'un *critère d'erreur* doit être défini. La détermination de ce critère reste relativement simple [29][31]. Les primitives étant positionnées de manière robuste, elles offrent généralement une bonne approximation de départ. Une autre simplification intéressante vient du fait que l'ajustement local ne nécessite pas l'emploi de la fonction exponentielle, ce qui constitue un gain précieux en temps de calculs. Nous proposons donc la fonction de mesure d'erreur suivante en chaque point  $P$  du contour pour la  $i^{\text{ème}}$  primitive :

$$\varepsilon_i(P) = f_i^2(\mathbf{D}_i^{-1}(P).P)$$

Équation 10

Taubin *et al.* [31] proposent une meilleure approximation de la distance définie par la fonction  $f_i$ , au sens où elle se rapproche de la distance euclidienne :

$$\delta_i(P) = \frac{|f_i(\mathbf{D}_i^{-1}(P).P)|}{\|\nabla f_i(\mathbf{D}_i^{-1}(P).P)\|} \approx \text{dist}(P, Z_i)$$

Équation 11

où  $\text{dist}$  représente la distance euclidienne entre le point  $P$  et la surface  $Z_i$ . Pour des raisons de stabilité, la fonction de l'Équation 10 est utilisée dans la méthode d'optimisation, et l'Équation 11 s'emploie pour fournir une mesure intuitive d'erreur une fois que le processus d'ajustement est terminé.

#### 2.4.6. Une méthode d'ajustement surfacique : la méthode tabou.

Dans notre situation, plusieurs obstacles sont présents :

- les équations décrivant les primitives sont non-linéaires (et fortement non linéaires si l'on considère les fonctions booléennes) ;
- les paramètres de déformations sont contraints et ne présentent pas les mêmes ordres de grandeur ;
- les points de contours peuvent ne pas être très nombreux, par exemple dans le cas du sous-échantillonnage, ou très éloignés les uns des autres si la densité d'échantillonnage varie [15] ;
- le modèle de déformation peut présenter des problèmes de continuité et de dérivabilité par exemple dans le cas de valeurs extrêmes.

Nous avons employé diverses méthodes comme *Levenberg-Marquardt* ou les *gradients conjugués* [12][38] (méthodes quasi-Newtonniennes), parfois avec succès, mais le plus souvent avec une grande instabilité. Pour tenter de résoudre les problèmes de divergence, nous avons implanté une méthode basée sur l'*homotopie*. Cette technique consiste à projeter les points de contour sur l'entité initiale, puis à les déplacer progressivement vers leur position réelle en relançant une optimisation numérique à chaque avancée. De cette manière la solution courante est plus « proche » de la solution finale à chaque étape, ce qui améliore la résistance aux extrema locaux ainsi qu'à la divergence. Nous avons observé une augmentation des temps de calcul de l'ordre d'un facteur assez proche de celui du nombre d'étapes intermédiaires mises en place, ainsi que l'apparition – beaucoup plus rare mais présente tout de même – de divergences. Enfin, la fonction de projection des points sur une primitive implicite est coûteuse.

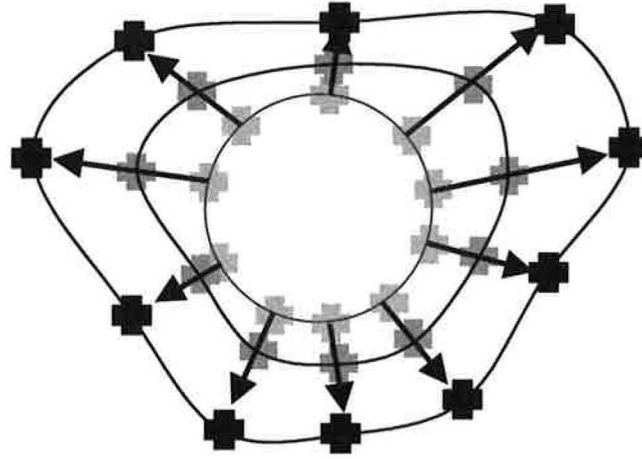


Figure 16

L'homotopie conduit les points de contour réels (en noir) de la forme de départ (cercle au centre) jusqu'à leur position finale. Une optimisation numérique est opérée à chaque étape pour déformer le cercle vers les nouveaux points (concentriques, du centre vers l'extérieur), ce qui renforce la robustesse de la méthode. Ce renforcement n'est malheureusement pas toujours suffisant et allonge de manière très sensible le temps de calcul.

Nous nous sommes alors tournés vers la méthode dite *tabou* [34][35][36][37], qui offre une alternative intéressante. Elle appartient à la famille des métaheuristiques, et permet de converger vers une solution proche de la solution idéale en évitant les extrema locaux. Basée principalement sur une recherche locale agressive sans utilisation de la dérivée, elle repose sur des principes génériques indépendants de tout contexte. Au terme de cette recherche, une méthode d'optimisation classique peut alors être employée avec succès pour préciser la solution.

#### 2.4.6.1. Introduction.

Un problème d'optimisation est défini par le couple  $(S, \zeta)$ , où  $S$  est l'ensemble des configurations possibles et  $\varepsilon$  est la fonction de coût. Pour notre ajustement surfacique,  $S$  représente l'ensemble de tous les vecteurs  $\mathbf{u}_i$  et  $\zeta$  est définie comme suit :

$$\zeta_i(B) = \frac{1}{n_B} \sum_{j=1}^{n_B} \varepsilon(\mathbf{u}_i, P_j)$$

Équation 12

où  $B$  est l'ensemble des points  $P_j$  de contour de la  $i^{\text{ème}}$  primitive, et  $n_B$  le nombre de ces points.

Nous appelons *voisinage* toute application  $N: S \rightarrow 2^S$  : à partir d'une configuration  $s$  de  $S$ , il est possible d'obtenir, via la fonction  $N$ , de nouvelles configurations « proches » de  $s$ . Cet ensemble noté  $N(s)$  appartient à l'ensemble des parties de  $S$ , noté  $2^S$ .



Une configuration  $s$  est dite *minimum local* pour le voisinage  $N$  si  $\forall s' \in N(s), \zeta(s) < \zeta(s')$ .

#### 2.4.6.2. Explications du fonctionnement de *tabou*.

Une méthode classique d'optimisation commence avec une solution initiale qui lui est donnée par :

$$s^0 \stackrel{def}{=} \mathbf{u}_i^0 = (t_x, t_y, t_z, \varphi, \theta, \psi, a, b, c, 0, \dots, 0)$$

Équation 13

où  $t_x, t_y, t_z$  désignent les coordonnées du centre de gravité de la classe,  $\varphi, \theta, \psi$  sont les angles d'Euler basés sur le trièdre d'inertie, et  $a, b, c$  sont les rayons ou valeurs propres de la matrice d'inertie. Le restant des paramètres est mis à zéro.

Une série de mouvements est réalisée, modifiant la valeur de la fonction  $\zeta$ . Le critère d'arrêt est habituellement un seuil sur  $\zeta$ . A l'itération  $t$ , et pour une configuration courante  $s$ , un voisinage  $N(s)$  est généré par perturbations de certaines coordonnées de la configuration  $s$ . L'amplitude de perturbation est bien sûr adaptable à la grandeur concernée, et peut être ajustée en fonction de la valeur de  $\zeta$  de manière à ne réaliser que de faibles variations lorsque l'algorithme est proche de la solution ou d'un certain seuil d'erreur. La méthode tabou choisit alors la meilleure configuration  $s' \in N(s)$ , **même si**  $\zeta(s') > \zeta(s)$ , ce qui permet d'éviter les minima locaux. Le nombre de voisins est en général assez faible ( $< 10$ ), ce qui réduit les évaluations.

Une dernière spécificité concerne la conservation des derniers mouvements effectués. Les dernières configurations parcourues sont stockées dans une liste appelée *liste tabou* et permettent de s'extraire d'un bouclage à court terme. La nature discrète des pas de mouvement permet de stocker dans la liste une valeur exacte de la configuration, ce qui évite des tests faussés par l'imprécision numérique. Les heuristiques de détermination de la taille de cette liste sont diverses, mais nous avons décidé de la fixer empiriquement à 10. Celle-ci est gérée classiquement comme une file (« dernier entré dernier sorti »).

L'algorithme global se déroule donc ainsi :

```

Choix d'une configuration initiale  $s_0$ 
NbMouvements  $\leftarrow 0$ 
 $s \leftarrow s^0$ 
 $s^* \leftarrow s$ 
Tant Que [(NbMouvements < NbMouvementsMax) Et ( $\zeta(s) > \zeta_{\min}$ )] Faire
    Générer  $N(s)$ 
    Choisir  $s' \in N(s)$  avec  $\zeta(s')$  minimal et  $s' \notin \text{ListeTabou}$ 
    Mettre  $s'$  dans ListeTabou
     $s \leftarrow s'$ 
    Si  $\zeta(s) < \zeta(s^*)$  Alors  $s^* \leftarrow s$  FinSi
    NbMouvements  $\leftarrow$  NbMouvements+1
FinTantQue

```

$s^*$  est la meilleure solution.

Algorithme 1

Une fois fixés, les paramètres de seuil et de perturbation ne requièrent plus aucune modification. Les heuristiques de perturbation, de taille de liste tabou, de mouvements ...etc. sont des aspects largement optimisables de notre méthode.

#### 2.4.7. Temps de Calcul.

Nous présentons dans le Tableau 3 les temps de reconstruction sur l'hamatum (en moyenne sur les cinq examens) et les deux fémurs.  $e_{\max}$  et  $e_{\text{moy}}$  désignent respectivement les erreurs maximales et moyennes en millimètres de la surface par rapport aux points de contour.

Objet	Primitives	Temps	$e_{\max}$	$e_{\text{moy}}$	Longueur API
Hamatum	4	4min	3.2mm	1.1mm	2cm
Hamatum	5	5min	2.8mm	0.8mm	2cm
Fémur	6	20min	5mm	2.4mm	40cm
Fémur	8	15min	4.7mm	2.0mm	40cm

Tableau 3

Ce tableau met en valeur la qualité obtenue avec un nombre faible de primitives. Le lecteur notera que la version du fémur avec 8 primitives se calcule plus rapidement, les classes convergeant vers des formes sphériques ce qui rend le cheminement de l'optimisation numérique plus rapide.

Les temps de calcul du Tableau 3 sont donnés pour une station de travail SGI O<sub>2</sub> avec 96 Mo de mémoire vive et un processeur MIPS R5000 à 180 Mhz. Le fémur est composé d'environ 25000 points de contour et l'hamatum d'environ 7500. Nous rappelons que le fémur mesure environ 40 cm de long, et l'hamatum 2 cm de long.

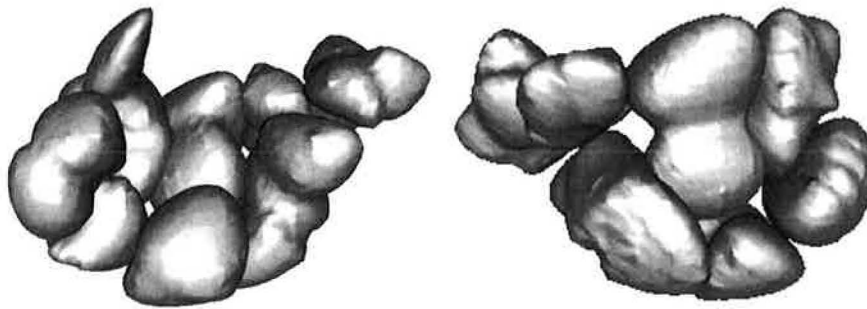


Figure 17

Figure 17 : différentes vues d'un carpe complet (huit os) obtenu par ajustement surfacique pour une erreur moyenne de l'ordre de 1mm. Le carpe complet mesure environ 6cm sur 4cm sur 3cm. Le facteur moyen de diminution est de 5 pour l'erreur moyenne, et de 2 pour l'erreur maximale, ceci étant dû principalement à la nature globale du modèle de déformation.

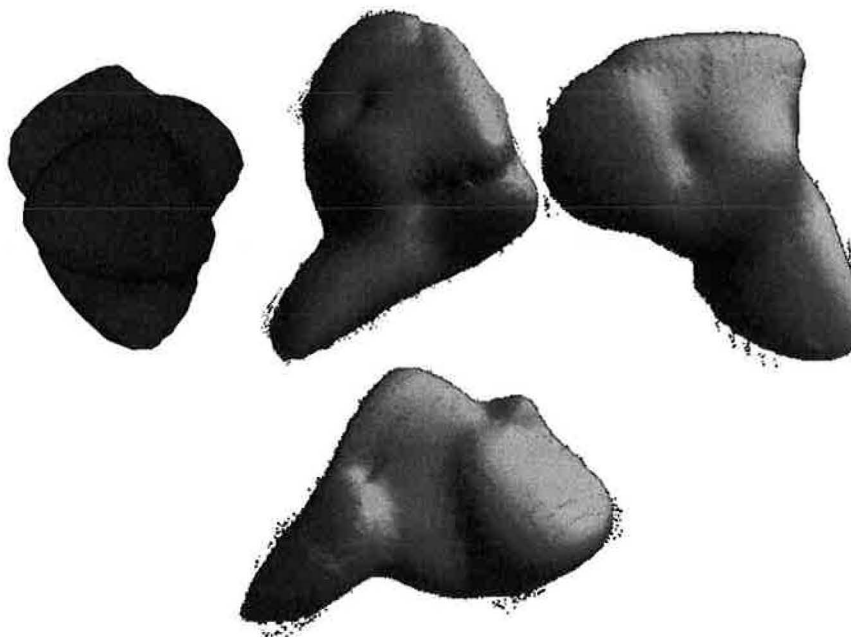


Figure 18

Figure 18 (de gauche à droite et de haut en bas) : détail du *trapeze* reconstruit, puis trois vues de l'*hamatum* avec superposition du nuage de points dans le module *Corpus 2000* (vision en temps réel).



Figure 19

Figure 19 : vue d'un foie tumoral après segmentation par filtrage 3D non-linéaire avec croissance de région hiérarchique 3D [39]. Le foie sain est en rouge, la lésion est en jaune. L'exportation au format VRML autorise la visualisation accélérée matériellement avec des effets de transparence par exemple comme c'est le cas ici.

#### 2.4.8. Récapitulatif de l'algorithme de reconstruction.

L'algorithme complet est le suivant :

Partitionnement des Points en  $k$  classes

Pour Chaque Classe Faire

    Sélectionner les points de contour de la classe

    Solution Initiale : superquadrique centrée au centre de gravité, orientée et dimensionnée selon les trois vecteurs principaux d'inertie.  $\varepsilon_1$  et  $\varepsilon_2$  sont mis à 1 (voir Équation 2). Quant à  $\vec{u}_i$ , il est initialisé comme indiqué dans l'Équation 13.

    Optimisation Numérique sur  $\varepsilon_1$  et  $\varepsilon_2$  (voir sections 2.4.5 et 2.4.6)

    Optimisation Numérique sur  $\vec{u}_i$  (voir sections 2.4.5 et 2.4.6)

FinPour

Algorithme 2

Optimisation Numérique sur tous les  $(\alpha_i, \beta_i)$  simultanément (voir section 2.4.4).

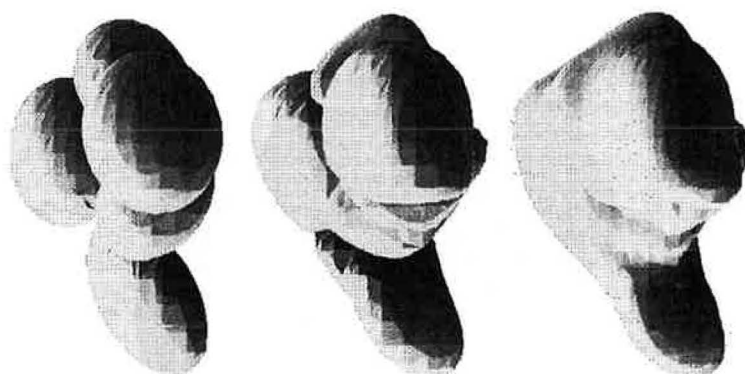


Figure 20

De gauche à droite (Figure 20) : solution initiale, déformation de chaque primitive et enfin ajustement de la fusion des primitives déformées.

### 3. REPRÉSENTATION GÉOMÉTRIQUE EXPLICITE.

Nous allons maintenant décrire la phase de discrétisation et d’affichage du E-Squelette, avec la gestion dynamique de la finesse de représentation en fonction du contexte d’étude. Cette section mettra en lumière les bénéfices de l’approche purement analytique et les considérations à propos de la transmission progressive d’un maillage tridimensionnel.



Figure 21

Figure 21 : cette vue (pied complet) avec superposition des points présente tous les avantages apportés par le E-Squelette : forme analytique de chaque os, niveau de détail et zoom sémantique dynamiques. Le pas de discrétisation varie en fonction de critères de distances ainsi que le nombre de primitives par os. Il n’y a pas de fonction de mélange.

#### 3.1. La polygonisation ou discrétisation.

Le processus de polygonisation de la surface se doit d’être rapide pour assurer l’interactivité. Nous avons décidé d’employer la méthode basée sur l’approximation linéaire par morceaux (*marching tetrahedra* [16][24][27]).

Pour éviter de parcourir l’espace 3D dans sa totalité (pour explication du pavage 3D, voir le chapitre traitant de la représentation tridimensionnelle), un point  $A$  est détecté aléatoirement à l’extérieur de la surface et un point  $B$  à l’intérieur de la surface. Le point de départ est ensuite déterminé par dichotomie sur la droite  $(AB)$ .

Nous n’avons pas développé de méthode adaptative [25] dans notre application, les objets étant suffisamment réguliers (car de nature organique) pour être discrétisés régulièrement. Durant la phase de polygonisation, l’objet est ramené à l’origine afin d’éviter au maximum tout dépassement dû à la fonction exponentielle. Enfin, la boîte englobante est ajustée au mieux en utilisant les données de taille et d’orientation des différentes classes.

Nos choix privilégiant la vitesse de discrétisation, nous avons été confronté néanmoins au problème classique des surfaces implicites : l’oubli d’une des composantes connexes [32][43]. Pour l’union exacte cela ne pose pas de difficulté, chaque primitive étant alors discrétisée séparément. Lorsque la peau implicite est mise en place (union booléenne),

l'arbre requiert d'être évalué en entier, ce qui provoque alors ce problème. Pour le contourner, nous avons développé une technique qui tire parti de la structure de l'arbre géométrique.

L'évaluation d'un nœud de l'arbre géométrique (et de tous ses descendants) n'est pas simplement réduit à un processus passif : lorsqu'un nœud de l'arbre renvoie une valeur en un point de l'espace, il conserve la valeur renvoyée la plus proche de la constante  $\epsilon$  (voir Équation 3). Après discrétisation de la composante connexe détectée par dichotomie, un parcours de l'arbre est effectué qui désactive les nœuds ayant renvoyé l'isovaleur  $\epsilon$  (ou une valeur très proche de celle-ci) et le processus est relancé jusqu'à désactivation complète. Cette technique évite la perte de composantes connexes due au fait de la détection de la surface par dichotomie.

L'algorithme est le suivant :

Tous les nœuds de l'arbre géométrique sont **activés**  
NombredeComposantesConnexes  $\leftarrow 0$

Répéter

    Trouver A à l'intérieur de l'objet

    Trouver B à l'extérieur de l'objet

    Trouver le point de  $(AB)$  sur la surface, par dichotomie

    NombredeComposantesConnexes  $\leftarrow$  NombredeComposantesConnexes+1

    Discrétiser la surface détectée en employant la méthode de Bloomenthal sur la racine de l'arbre géométrique

    Désactiver les nœuds de l'arbre ayant renvoyé la valeur  $\epsilon (+/-\epsilon)$

Jusqu'à totale désactivation

Renvoyer NombredeComposantesConnexes

Algorithme 3

En sortie le flot de polygones est envoyé à l'accélérateur graphique, enrichi d'une indication pour l'utilisateur du nombre de composantes connexes détectées. Le lecteur notera que cette information topologique peut-être fausse pour deux raisons essentielles :

- le pas de discrétisation de l'espace est trop grand, ce qui peut entraîner la fusion de deux composantes connexes ;
- le processus de détection d'une composante connexe de la surface échoue. Ce problème survient rarement en pratique, étant donné l'encadrement au plus juste de l'entité tridimensionnelle.

### 3.2. Zoom sémantique et niveaux de détails.

Nous montrons à présent les possibilités de rendu à la volée de la représentation géométrique par E-Squelette.

#### 3.2.1. Le zoom sémantique (ZS).

C'est l'heuristique de choix du nombre de classes à afficher. Bien entendu, ce nombre est lié à la distance de vision ainsi qu'au point d'attention : si la zone explorée est très localisée, l'algorithme va détecter quelles sont les classes observées et produire ainsi un découpage plus prononcé sur cette zone :

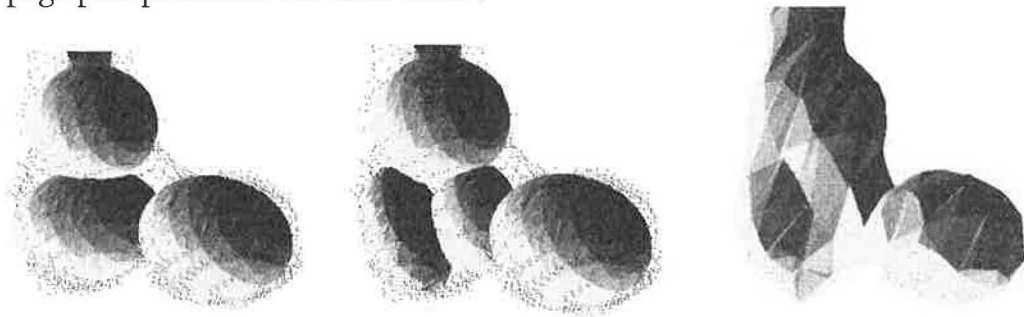


Figure 22

Figure 22 : lorsque l'étude se focalise sur un point précis, un raffinement peut être opéré localement. Nous montrons ici le recalcul dynamique pour la tête du fémur, sans peau implicite afin de mettre en évidence l'ajout d'une classe. L'image la plus à droite est la version avec peau implicite qui est effectivement affichée.

Quant au nombre de classes pour la vision globale, nous avons une heuristique définie comme suit :

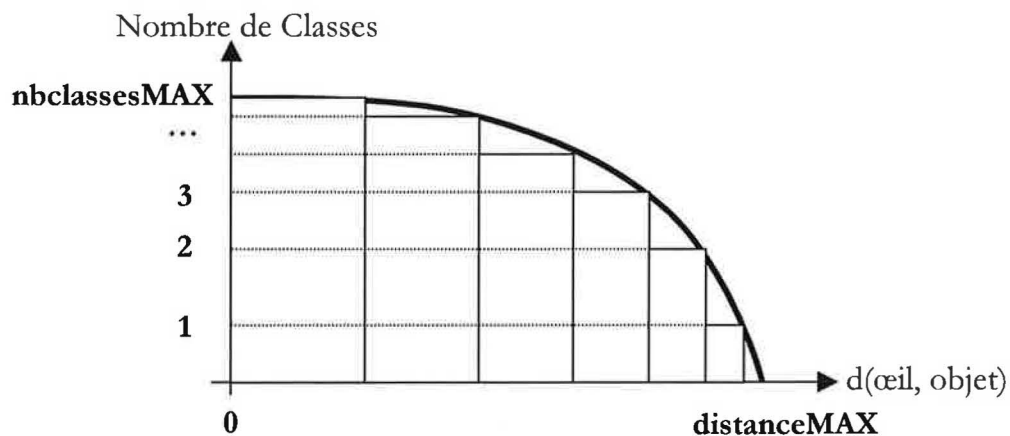


Tableau 4

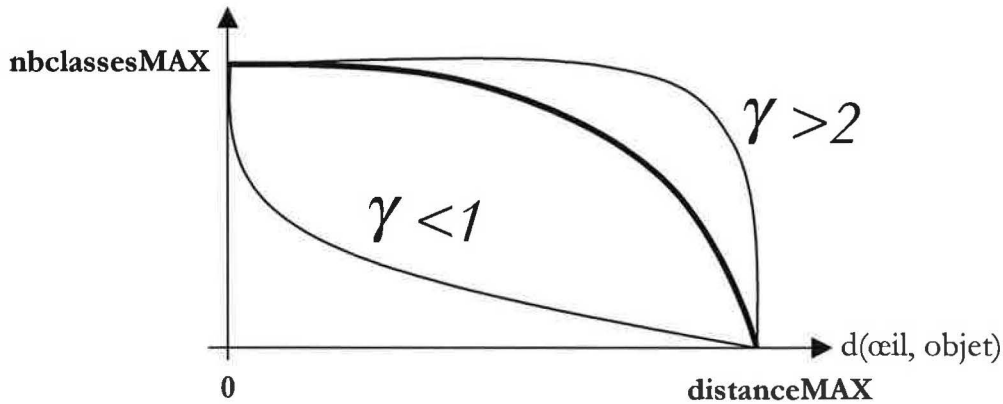


Nous proposons donc l'équation suivante pour déterminer le nombre de classes conformément au Tableau 4 :

$$nbclasses = \left\lceil nbclassesMAX \sqrt[\gamma]{1 - \left( \frac{dist(oeil, objet)}{distanceMAX} \right)^\gamma} \right\rceil$$

Équation 14

où  $\gamma$  régule ainsi la courbure :



Équation 15

La motivation principale est de décimer rapidement si l'objet est « éloigné » (relativement à sa taille), et d'autoriser moins de variations à faible distance afin d'éviter des phénomènes d'apparitions soudaines de détails. Bien entendu la taille de l'objet peut être prise en considération et pourra pondérer la fonction présentée dans l'Équation 14.

### 3.2.2. Les niveaux de détails.

Le pas de discrétisation est également régulé dynamiquement en fonction de la distance de vision. Des contraintes liées à la puissance matérielle peuvent être prises en compte au moyen de tests de performance préliminaires.

De la même manière que dans la section 3.2.1, le pas de discrétisation est calculé comme suit :

$$pas = \left\lceil tailleOBJET \sqrt[\gamma]{1 - \left( \frac{dist(oeil, objet)}{distanceMAX} \right)^\gamma} \right\rceil$$

Équation 16

où  $tailleOBJET$  représente par exemple la longueur de l'axe principal d'inertie de l'objet entier (ou de la classe dans le cas d'un affichage classe par classe sans peau implicite).

### 3.2.3. Combinaison Dynamique.

Combiner les deux approches offre un puissant outil pour la visualisation interactive. Il devient en effet assez simple de garantir un taux d'images par seconde élevé en simplifiant la représentation en temps réel. Il reste toutefois à améliorer le processus d'évaluation de l'arbre géométrique de manière à accélérer la production de polygones. C'est cette phase qui constitue le goulot d'étranglement puisque la chaîne graphique matérielle combinée à une régulation constante du flot de polygones est aujourd'hui largement capable d'afficher un grand nombre de polygones. Nous en venons ainsi à la situation paradoxale où contrôler trop souvent le maillage ralentit fortement le rendu interactif. Pour cette raison, nous avons défini des plages de maillage constant assez larges afin d'éviter une remise en cause trop pénalisante, ainsi qu'un algorithme de recalcul qui est le suivant :

Tant Que Vrai Faire

    Calculer Pas de Discrétisation en fonction de la distance œil/objet

    Calculer Nombre de Classes en fonction de la distance œil/objet

    Si Changement par rapport à la situation courante Alors

        Si le Fichier existe déjà sur le Disque Alors

            Charger le Maillage

        Sinon

            Discrétiser le nouveau Maillage

            Sauvegarder le nouveau Maillage

        FinSi

    FinSi

    Afficher le Maillage

FinTantQue

Algorithme 4

Le diagramme suivant précise visuellement l'effet de la régulation zoom sémantique/niveaux de détail :

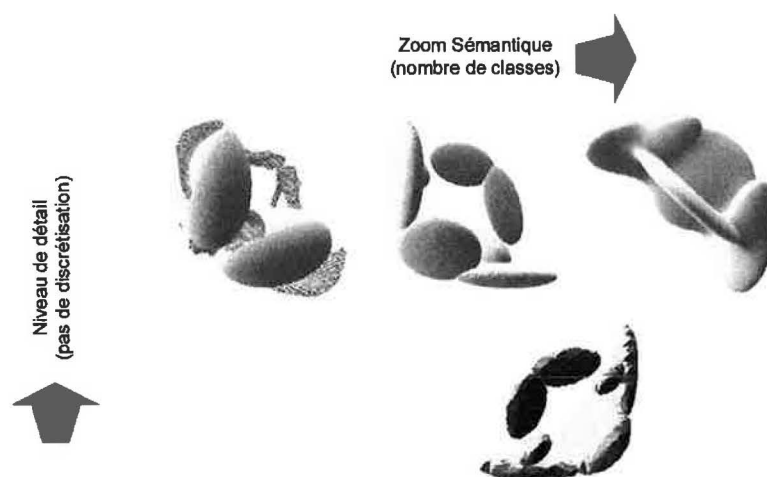


Figure 23

Figure 23 : combinaison du zoom sémantique et du niveau de détail (pour l'os iliaque).

Voici un exemple de visualisation interactive complète :

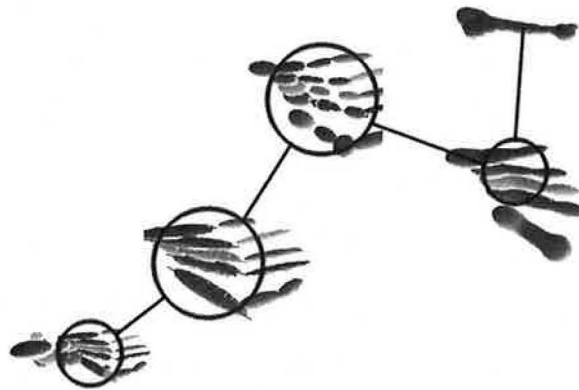


Figure 24

La Figure 24 démontre la faculté d'adaptation de la chaîne de visualisation par E-Squelette, avec comme niveau maximal de finesse la reconstruction par ajustement surfacique. Cette dernière phase est bien sûr effectuée en précalcul (ou en tâche de fond), et ne peut en aucun cas être opérée à la volée.

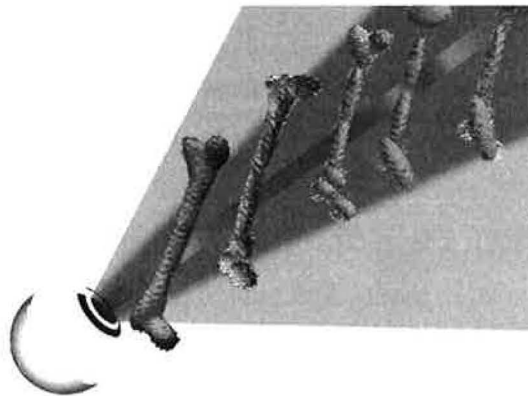


Figure 25

Figure 25 : principe de la vision dynamique (cas du fémur), avec augmentation progressive de la qualité du maillage. L'objet le plus proche de l'œil est obtenu par ajustement surfacique, il représente le niveau de zoom sémantique le plus élevé mais reste candidat à la régulation du pas de discrétisation du fait de sa nature analytique.



Figure 26

Figure 26 : utilisation du module E-Squelette (fenêtre en bas à droite) dans *Corpus 2000*. Le lecteur remarquera la vision filaire du foie, les maquettes pouvant être dégradées automatiquement pour fluidifier l'affichage en cas d'absence d'accélération matérielle.

### 3.3. Compression de Données et Transmission Progressive par Réseau.

Le choix de la structure de données autorise une représentation très compacte de l'objet transformé en E-Squelette. A partir de l'arbre hiérarchique, n'importe quelle représentation géométrique peut être générée localement. La reconstruction nécessitant un ajustement surfacique reste compacte du fait de l'expression analytique de la surface, même déformée, et peut de surcroît être précalculée.

Nous avons évalué le taux de compression d'un arbre géométrique par rapport à une représentation polygonale :

- par rapport à un maillage de précision moyenne : 1 / 60
- par rapport à un maillage très fin : 1 / 100.

Nous avons également développé une passerelle vers le format VRML (*Virtual Reality Modeling Language*), en intégrant la notion de niveaux de détails et zoom sémantique dynamiques, puisqu'elle est supportée par ce format.

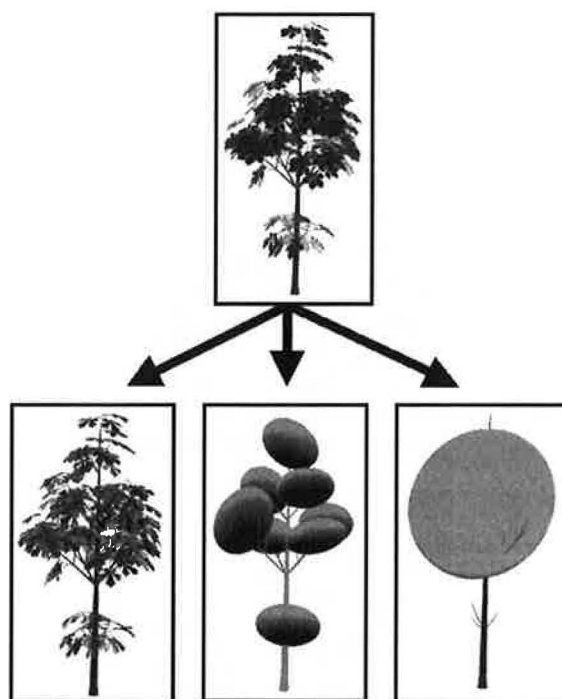


Figure 27

Figure 27 : génération au format VRML d'un végétal. La version complète est présentée en haut, tandis qu'en bas sont produits différents niveaux de détails de la couronne. L'aspect hiérarchique des enveloppes englobantes autorise l'accélération des calculs en lancer de rayons [41].

L'aspect totalement analytique de l'arbre géométrique autorise des sorties vers le lancer de rayon, comme nous le montre cette image d'un hamatum calculée avec POVray :



Figure 28

#### **4. CONCLUSION.**

Nous avons montré dans ce chapitre la grande flexibilité du modèle de E-Squelette en terme de production de représentation géométrique. Il semble plus important toutefois de proposer une information affinée selon le contexte qu'une représentation géométrique très fine. Même si le E-Squelette permet d'atteindre ce degré de précision, c'est au prix d'un coût calculatoire élevé. Dans le champ applicatif médical, l'information la plus précieuse est celle qui traite de la nature interne des tissus [14][20], le support visuel n'étant présent que pour guider l'observation ou encore à des fins de démonstration.

Néanmoins, nous avons voulu créer un modèle de représentation très large, pouvant convenir à une multitude d'applications et ne présumant pas de la nature des objets à reconstruire. Nous discuterons des limitations dans la conclusion de ce mémoire, montrant ainsi que malgré nos investigations, de nombreuses voies restent encore à explorer.



## 5. BIBLIOGRAPHIE.

- [1] E. Castillo and A. Iglesias, « *Some Characterizations of Families of Surfaces Using Functional Equations.* », ACM Transactions on Graphics, vol. 16, No. 3, pp. 296-318, July 1997
- [2] J. F. Blinn, « *A Generalization of Algebraic Surface Drawing.* », ACM Transactions on Graphics, vol. 1, No. 3, pp. 235-256, July 1982
- [3] F. Banégas, D. Michelucci, M. Roelens and M. Jaeger, « *Automatic Adaptive Surface Reconstruction from Ellipsoidal Skeleton.* », 1999, The Fourth International Workshop on Implicit Surfaces '99, Bordeaux, France.
- [4] F. Banégas, D. Michelucci, M. Roelens and M. Jaeger, « *Automatic Extraction of Significant Features from 3D Point Clouds by Ellipsoidal Skeleton. Applications in Vision and Geometrical Characterization.* », 1999, International Conference in Visual Computing '99, Goa, India.
- [5] B. Wyvill, G. Wyvill, « *Field Functions for Implicit Surfaces.* », The Visual Computer, 5, pp. 75-82, Springer-Verlag, 1989
- [6] D. Rossin, M-P. Cani-Gascuel, « *Plateforme d'Expérimentation pour la Modélisation par Surfaces Implicites.* », Journées « Modeleurs Géométriques », Grenoble, 17-19 septembre 1997
- [7] A. Pasko, V. Adzhiev, A. Sourin and V. Savchenko, « *Function Representation in Geometric Modeling : Concepts, Implementation and Applications.* », The Visual Computer, 11, pp. 429-446, Springer-Verlag, 1995
- [8] J. Bloomenthal, K. Shoemake, « *Convolution Surfaces.* », SIGGRAPH'91, Computer Graphics, vol. 25, No. 4, pp. 251-256, July 1991
- [9] T. Duff, « *Interval Arithmetic and Recursive Subdivision for Implicit Functions and Constructive Solid Geometry.* », SIGGRAPH'92, Computer Graphics, vol. 26, No. 2, pp. 131-138, July 1992
- [10] M. Kisters, « *An Extension of the Potential Method to Higher-Order Blendings.* », SIGGRAPH'91, Computer Graphics, vol. 25, No. 4, pp. 329-337, July 1991
- [11] H. Hoppe, « *Progressive Meshes.* », SIGGRAPH'96, pp. 99-108, 1996.
- [12] R. Fletcher, « *Conjugate Gradient Methods for Indefinite Systems.* », Numerical Analysis Dundee, Springer-Verlag, 1995
- [13] A. Barr, « *Global and Local Deformations of Solid Primitives.* », Computer Graphics, vol. 18, pp. 21-30, ACM Press, 1984
- [14] V. Burdin, C. Roux, C. Lefèvre, E. Stindel, « *Modeling and Analysis of 3D Elongated Shapes with Applications to Long Bone Morphometry.* », IEEE Transactions on Medical Imaging, vol. 15, pp. 79-91, IEEE, February 1996
- [15] G. Danuser, M. Stricker, « *Parametric Model Fitting : From Inlier Characterization to Outlier Detection.* », IEEE Transactions on PAMI, vol. 20, pp. 263-280, IEEE, March 1998
- [16] W. E. Lorensen, H. E. Cline, « *Marching Cubes : A High Resolution 3D Surface Algorithm.* », Computer Graphics, vol. 21, pp. 163-169, ACM Press, July 1987
- [17] A. Leonardis, A. Jaklic and F. Solina, « *Superquadrics for Segmenting and Modeling Range Data.* », IEEE PAMI, vol. 19, pp. 1289-1295, IEEE, November 1997

- [18] C. T. Lim, G. M. Turkiyyah, M. A. Ganter and D. W. Stortri, « *Implicit Reconstruction of Solids from Cloud Point Sets.* », Solid Modelling'95, pp. 393-402, ACM Press, 1995
- [19] S. Muraki, « *Volumetric Shape Description of Range Data using Blobby Model.* », Computer Graphics, vol. 25, pp. 227-235, ACM Press, 1991
- [20] M. L. Rhodes, « *Computer Graphics in Medicine : A Complex Partnership.* », IEEE Computer Graphics and Applications, pp. 22-28, IEEE, January-February 1997
- [21] S. Sclaroff, A. Pentland, « *Generalized Implicit Functions for Computer Graphics.* », vol. 25, pp. 247-250, ACM Press, July 1991
- [22] V. V. Savchenko, A. A. Pasko, O. G. Okunev and T. L. Kunii, « *Function Representation of Solids Reconstructed from Scattered Surface Points and Contours.* », Eurographics'95, Blackwell Publishers, 1995
- [23] R. Durikovic, K. Kaneda and H. Yamashita, « *Reconstructing a 3D Structure with Multiple Deformable Solid Primitives.* », Computer and Graphics, vol. 21, pp. 611-622, Pergamon, 1997
- [24] J. Bloomenthal, « *An Implicit Surface Polygonizer.* », Graphics Gems IV, Paul Heckbert, Academic Press, New York, 1994
- [25] M. Hall, J. Warr, « *Adaptive Polygonization of Implicitly Defined Surfaces.* », IEEE Computer Graphics and Applications, IEEE Computer Society Press, vol. 10, pp. 33-42, November 1990
- [26] D. Metaxas, D. Terzopoulos, « *Dynamic Deformation of Solid Primitives with Constraints.* », Proceedings of SIGGRAPH'92, vol. 26, ACM SIGGRAPH, pp. 309-312, July 1992
- [27] G. Nielson, B. Hamann, « *The Asymptotic Decider : Resolving the Ambiguity in Marching Cubes.* », Proceedings of Visualization'91, Nielson G. M., Rosenblum L. Eds., pp. 83-91, october 1991
- [28] G. D. Purvis, C. Culberson, « *On the Graphical Display of Molecular Electrostatic Force-Fields and Gradients of the Electron Density.* », NATO Conference, pp. 317-332, July 1985
- [29] V. Pratt, « *Direct Least-Squares Fitting of Algebraic Surfaces.* », Proceedings of SIGGRAPH'87, vol. 21, ACM SIGGRAPH, pp. 145-152, 1987
- [30] A. Ricci, « *A Constructive Geometry for Computer Graphics.* », The Computer Journal, vol. 16, pp. 157-160, May 1973
- [31] G. Taubin, « *Distance Approximation for Rasterizing Implicit Curves.* », ACM Transactions on Graphics, vol. 13, pp. 3-42, 1994
- [32] B. Wyvill, K. van Overveld, « *Polygonization of Implicit Surfaces with Constructive Solid Geometry.* », Journal of Shape Modeling, vol. 2, World Scientific Publishing, 1997
- [33] G. Wyvill, « *Data Structures for Soft Objects.* », The Visual Computer, vol. 2, pp. 227-234, August 1986
- [34] J-K. Hao, P. Galinier and M. Habib, « *Métaheuristique pour l'Optimisation Combinatoire et l'Affectation sous Contraintes.* », Research Report, EMA-EERIE
- [35] J-K. Hao, R. Dorne, « *Empirical Studies of Heuristic Local Search for Constraint Solving.* », Proceedings of CP-96, LNCS 1118, pp. 194-208, Cambridge, MA, USA, 1996
- [36] F. Glover, « *Tabu Search ; Part I.* », ORSA Journal on Computing, vol. 1, No. 3, pp. 190-206, 1989

- [37] F. Glover, « *Tabu Search ; Part II.* », ORSA Journal on Computing, vol. 2, No. 1, pp. 4-32, 1989
- [38] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, « *Numerical Recipes in C : The Art of Scientific Computing Second Edition.* », Cambridge University Press, 1988-1992
- [39] S. Chemouny, H. Joyeux, B. Masson, F. Borne, M. Jaeger and O. Monga, « *Advanced 3D image processing techniques for liver and hepatic tumors location and volumetry.* », Medical Imaging 1999 - Image Processing, K.M. Hanson (Ed.), San Diego, USA, SPIE The International Society for Optical Engineering, SPIE Proceedings Series (3661-1), pp. 761-771, 20-26/02/1999
- [40] Z. Ka, I. Alesi, « *Blending Techniques for Procedural Implicit Surfaces.* », master's thesis, Department of Computer Science, the University of Calgary, Alberta, 1991
- [41] J. Dautzat, F. Sillion, C. Nougier and C. Damez, « *Specific methods for radiative transfer simulations on multi-scal plant mock-ups.* », Second International workshop on Functional-Structural Tree Models, Clermont-Ferrand, France, 1998
- [42] Hitoshi Nishimura, Makoto Hirai, Toshiyuki Kawai, Toru Kawata, Isao Shirakawa and Koichi Omura, « *Object Modeling by Distribution Function and a Method of Image Generation* », Transactions of the Institute of Electronics and Communication Engineers of Japan, 1985, vol. J68-D, no. 4, pp. 718-725, in Japanese (English translation by Takao Fujiwara, Advanced Studies in Computer Aided Art and Design, Middlesex Polytechnic, England, 1989)
- [43] M. Desbrun, N. Tsingos and M-P. Gascuel, « *Adaptive Sampling of Implicit Surfaces for Interactive Modeling and Animation.* », Computer Graphics Forum, December 1996

# **7. Reconnaissance de Formes par E-Squelette.**

## **1. INTRODUCTION.**

Le terme « reconnaissance de forme » désigne l'étude de la manière dont les machines observent l'environnement et apprennent à distinguer les entités qui le composent. Malgré presque cinquante années de recherches dans ce domaine, le problème général de l'automatisation n'est toujours pas résolu. Pourtant le besoin d'identification automatique et de classification de l'information se fait de jour en jour de plus en plus présent : bases de données d'information, reconnaissance de la parole et de l'écriture...Etc.

Nous présentons dans ce chapitre une technique permettant de classer, comparer et reconnaître les solides rigides tridimensionnels [1]. Ses fondements sont en partie tirés du vivier des techniques classiques, sur lesquels s'appuient des techniques originales et adaptées au problème qui nous concerne.

## 2. RAPPEL DE LA DÉFINITION FORMELLE DU E-SQUELETTE.

Le lecteur se reportera au chapitre *Le E-Squelette* pour un développement complet.

On définit un *objet tridimensionnel*  $O$  par un ensemble de points  $P_r$  de l'espace représentés par les quadruplets :

$$(x_r, y_r, z_r, \omega_r)$$

où  $x_r$ ,  $y_r$  et  $z_r$  sont les coordonnées dans  $\mathbb{R}^3$  de  $P_r$  et  $\omega_r$  leur pondération dans  $\mathbb{R}$ .

Pour un *niveau sémantique*  $n$ , le *squelette ellipsoïdal* (où *E-squelette*)  $\xi_n$  d'un objet tridimensionnel  $O$  est défini par l'ensemble :

$$\xi_n \stackrel{def}{=} \{C_1, \dots, C_{n+1}\}$$

Équation 1

où  $C_i (i \in \{1, \dots, n+1\})$  est une classe caractérisée par le vecteur  $(\alpha_1^i, \dots, \alpha_p^i)$  portant l'ensemble de paramètres  $\alpha_j^i \in \mathbb{R}^{k_j}$ , avec  $k_j$  entier naturel. Remarquons que  $\alpha_j^i$  peut aussi être une valeur discrète.

L'ensemble présenté dans l'Équation 1 est la clé de la technique que nous avons développée pour la reconnaissance automatique de formes.

### 3. LE PIPELINE DE MESURE.

Le pipeline de mesure décrit un processus que nous avons mis au point permettant de « mesurer » une « différence » entre des E-Squelettes, c'est-à-dire fournir une valeur de ressemblance entre ceux-ci. Comme nous allons le découvrir, le pipeline est bâti selon une philosophie de modularité et d'évolutivité.

#### 3.1. Pourquoi un pipeline ?

Le pipeline tente d'apporter une réponse à la question : « *quel est le degré de ressemblance (ou de dissemblance) entre deux objets à cette distance de vision et selon quels critères ?* ».



Figure 1

Les objets présentés en Figure 1 se ressemblent beaucoup sous certains aspects (nature, taille...), mais sont différents sous d'autres (couleur, forme...). Obtenir un indice de similarité combinant les critères désirés est une solution pour classer, comparer et reconnaître tout type d'objet : c'est le but du pipeline adapté au E-Squelette.

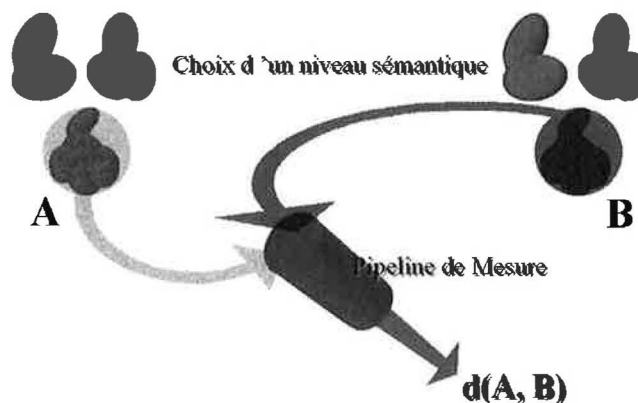


Figure 2

Dans la Figure 2, les éléments **A** et **B** sont préalablement transformés en E-Squelettes, **selon la même méthode**. Un niveau sémantique – et donc un nombre de classes – est ensuite choisi, ainsi que les paramètres à prendre en compte dans la mesure de ressemblance. En sortie du pipeline, la valeur **d(A, B)** est retournée.

### 3.2. Extraction de paramètres caractéristiques : les fonctions d'extraction.

Les données contenue dans  $\xi_n$  peuvent être obtenues à deux stades différents dans le processus de représentation par squelette ellipsoïdal :

- durant la phase de partitionnement, au travers de *fonctions d'extractions*  $\phi_j$  ;
- après la phase de partitionnement, en employant des *fonctions de synthèse* qui ont comme paramètres les données obtenues par l'emploi des fonctions d'extraction.

### 3.3. Fonctionnement du pipeline de mesure.

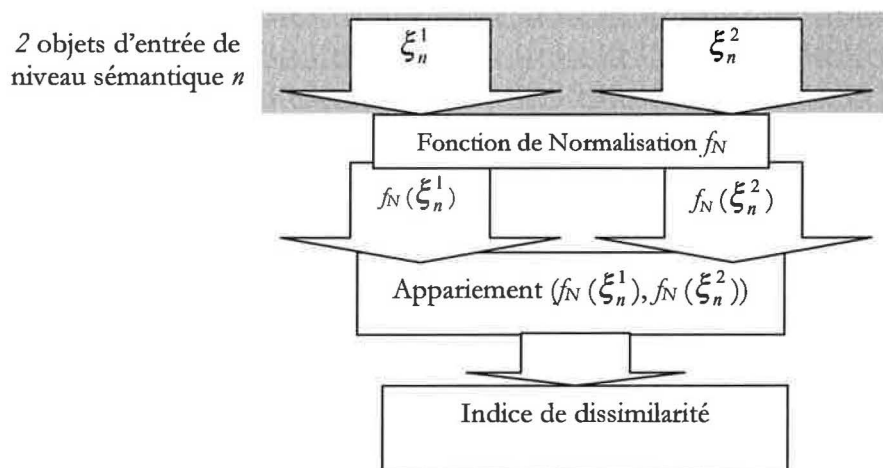


Figure 3

Les données provenant des deux E-Squelettes de niveau sémantique  $n$  sont transformées en premier lieu par la *fonction de normalisation*  $f_N$ . Un processus d'*appariement optimal* est ensuite exécuté pour déterminer l'indice de dissimilarité du couple de E-Squelettes.

#### 3.3.1. La fonction de normalisation $f_N$ .

Selon les besoins, toute fonction de normalisation peut être intégrée en phase de pré-appariement.



### 3.3.2. L'Appariement optimal.

Soient deux objets  $A$  et  $B$  décrits par leurs E-squelettes de niveau sémantique  $n$ , que nous noterons  $\mathcal{A}$  et  $\mathcal{B}$  pour des raisons de clarté :

$$\begin{cases} A = \{(\alpha_1^1, \dots, \alpha_p^1), \dots, (\alpha_1^{n+1}, \dots, \alpha_p^{n+1})\} \\ B = \{(\beta_1^1, \dots, \beta_p^1), \dots, (\beta_1^{n+1}, \dots, \beta_p^{n+1})\} \end{cases}$$

Équation 2

$\mathcal{A}$  et  $\mathcal{B}$  sont tous deux *ensembles* de  $n+1$  classes. Chaque classe est à son tour décrite par un vecteur-paramètre  $\alpha_j^i \in \mathbb{R}^k$  (à valeur continue ou discrète). Ce vecteur est calculé à partir des données-échantillons de l'objet concerné, grâce à une fonction d'extraction  $\phi_j$ . Il contient de manière élémentaire les valeurs propres et vecteurs propres de la matrice d'inertie de la classe concernée, mais peut être étendu (densité moyenne, volume exact, relations d'adjacence...etc.). Rappelons qu'il peut avoir subi une phase de normalisation au préalable. A tout  $\alpha_j \in \mathbb{R}^k$  est associé une *fonction de distance*  $\delta_j(\mathbb{R}^k \times \mathbb{R}^k) \rightarrow \mathbb{R}$  vérifiant les propriétés usuelles des métriques.

Le problème consiste à déduire de  $\mathcal{A}$  et de  $\mathcal{B}$  la distance  $D(\mathcal{A}, \mathcal{B})$ . Une définition naïve serait la suivante :

$$D_{fausse}(\mathcal{A}, \mathcal{B}) = d(A_1, B_1) + d(A_2, B_2) + \dots + d(A_{n+1}, B_{n+1})$$

Équation 3

où  $A_i$  et  $B_i$  sont respectivement les classes de  $\mathcal{A}$  et de  $\mathcal{B}$ . Cet appariement implicite est bien sûr faux étant donné que  $\mathcal{A}$  et  $\mathcal{B}$  sont des ensembles et non des vecteurs : l'ordre des classes est absent. Pour résoudre le problème, il faut déterminer le « bon appariement » entre  $\mathcal{A}$  et  $\mathcal{B}$ .

Un appariement est une correspondance biunivoque (ou bijection) entre deux ensembles de même cardinalité. Notons  $app(A_i)$  la classe  $B_j$  avec laquelle  $A_i$  est appariée. La distance  $D(\mathcal{A}, \mathcal{B})$  est alors :

$$D(\mathcal{A}, \mathcal{B}) = d(A_1, app(A_1)) + d(A_2, app(A_2)) + \dots + d(A_{n+1}, app(A_{n+1}))$$

Équation 4

Il reste à définir le bon appariement : c'est celui, parmi les  $(n+1)!$  appariements possibles, qui minimise la fonction de l'Équation 4. Ce problème classique d'appariement optimal se réduit à la recherche du *flot maximum de coût minimum* dans un graphe. Pour le résoudre, nous avons utilisé la méthode de Roy optimisée exposée en [17], qui est en  $O(n^3)$  ; nous n'avons pas trouvé de méthode plus efficace dans la littérature, ni de preuve d'optimalité. Pour un développement complet de la méthode, le lecteur se reportera à la section 3.4.

Cette technique d'appariement est également employée pour la mise en correspondance géométrique d'objets 3D [8], l'interpolation pour le *morphing* 3D [18] et la mesure de distance entre graphes valués [9].

### 3.3.3. Distance entre deux classes.

Donnons une définition possible pour la distance entre une classe  $a$  du E-Squelette  $A$  et une classe  $b$  du E-Squelette  $B$ . Soient  $\alpha_1$  et  $\alpha_2$  les deux plus grandes valeurs propres de la matrice de dispersion de la classe  $a$  et  $\beta_1, \beta_2$  les deux plus grandes valeurs propres de la matrice de dispersion de la classe  $b$ . Voici un exemple de distance possible entre  $a$  et  $b$  :

$$d(a, b) = |\alpha_1 - \beta_1| + |\alpha_2 - \beta_2|$$

Équation 5

(nous admettons que les valeurs propres sont positives, ce qui est le cas pour la matrice de dispersion).

Pour les comparaisons menées dans la suite de ce chapitre, nous avons employé une distance intégrant à la fois les valeurs propres des classes et la position des centres de gravité de chaque classe exprimée dans le repère de chaque objet. D'autres paramètres ont été intégrés (orientation de l'axe principal de chaque classe par exemple) mais ils n'ont pas apporté d'information supplémentaire lors de la reconnaissance, aussi avons-nous privilégié la solution la plus simple.

### 3.3.4. La matrice de distance $\mathbf{M}_d$ .

Lorsque plus de deux E-Squelettes sont injectés dans le pipeline de mesure, c'est la matrice  $\mathbf{M}_d$  qui est alors produite en sortie et qui contient tous les indices de dissimilarité entre chaque couple de E-Squelettes. C'est une matrice carrée de dimension  $m$  ( $m$  E-Squelettes injectés), symétrique, dont les valeurs diagonales sont nulles si la propriété d'identité des métriques est respectée.

## 3.4. Implantation de l'appariement optimal.

L'appariement optimal permet de calculer la valeur de l'indice de dissimilarité entre deux E-Squelettes représentés par des ensembles de classes. La méthode naïve permettant de tester toutes les permutations possibles entre les  $n+1$  classes possède une complexité en  $O(n!)$ , ce qui l'exclut d'office pour  $n > 10$ .

Pour contourner le problème, nous avons employé la technique dite du *graphe de flot maximal et de coût minimal* [3].

### 3.4.1. Initialisation de l'algorithme.

Soit  $\mathcal{A}$  le E-Squelette contenant les classes  $a_1 \dots a_{n+1}$  et  $\mathcal{B}$  le E-Squelette contenant les classes  $b_1 \dots b_{n+1}$ . Pour calculer l'appariement optimal entre  $\mathcal{A}$  et  $\mathcal{B}$  nous construisons un graphe orienté dont les arcs sont valués (Figure 4) :

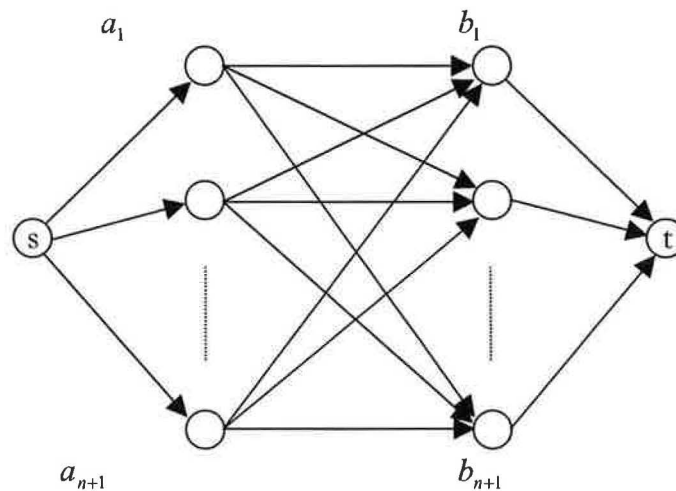


Figure 4

Les caractéristiques d'un tel graphe sont les suivantes :

- chaque sommet représente une classe ;
- chaque sommet décrivant le premier E-Squelette est relié à tous les sommets décrivant le second ;
- la source  $s$  est reliée à toutes les classes du premier E-Squelette. Le puits  $t$  est relié à toutes les classes du second ;
- chaque arc possède une capacité minimale nulle et une capacité maximale égale à 1 (ce sont les bornes de la valeur de flux transitant le long de cet arc) ;
- le flux admis pour chaque arc est à valeur entière. Avec l'hypothèse formulée précédemment, sa valeur est donc soit 0, soit 1 ;
- le coût d'un arc est la distance entre les deux classes (voir section 3.3.3) représentées par les deux sommets de l'arc. Tous les coûts sont obligatoirement positifs.

Une fois ce graphe construit, l'algorithme classique de flot maximal et de coût minimal est applicable. Il est présenté dans la section suivante (3.4.2).

### 3.4.2. L'algorithme de maximisation de flux pour un coût minimal.

A chaque étape, un *graphe d'écart* est construit comme suit :

Chaque arc  $u \rightarrow v$  du graphe initial (voir Figure 4) de capacité maximale  $m$  et traversé par un flot  $0 \leq f \leq m$ , génère dans le graphe d'écart :

- un arc  $u \rightarrow v$  de capacité maximale  $m - f$ , quand  $f < m$  ; en d'autres termes, il est possible de faire passer davantage de flot de  $u$  à  $v$ . Le coût de cet arc est celui de l'arc  $u \rightarrow v$  du graphe initial ;
- un arc  $v \rightarrow u$  de capacité  $f$ , quand  $f > 0$  ; autrement dit il est possible de faire passer du flot de  $v$  vers  $u$ , en diminuant le flot actuel  $f$  dans  $u \rightarrow v$ . Le coût de cet arc  $v \rightarrow u$  est l'opposé de celui de l'arc  $u \rightarrow v$  dans le graphe initial.

Nous avons choisi d'employer la méthode de Roy [17]. Celle-ci part d'un flot nul, et à chaque étape, sature le chemin de coût minimal dans le graphe d'écart courant : le couplage courant compte donc successivement  $0, 1, 2, \dots, n+1$  arcs et représente l'un des couplages de coût minimal ayant cette cardinalité.

Remarque :

- Les flux sont entiers à toutes les itérations, ce qui assure que le flux résultant est entier.

La complexité de la méthode originelle de Roy est en  $O(|V|^2|E|)$ , où  $V$  est l'ensemble des sommets du graphe et  $E$  l'ensemble des arêtes du graphe. Pour un niveau sémantique  $n$  nous avons donc une complexité en  $O(n^4)$  - le graphe étant complet - ce qui n'est pas optimal. Le graphe d'écart présentant des coût négatifs dans les arcs inverses, l'algorithme de Dijkstra de calcul de chemin de coût minimal ne peut pas être utilisé. Ceci est regrettable, d'autant plus qu'une telle méthode est optimale pour un graphe dense ( $O(n^2)$ ), ce qui donnerait une meilleure complexité en  $O(n^3)$ . Une astuce [17] existe toutefois, que nous avons implanté dans le module de comparaison. Nous avons trouvé dans la littérature [15] un algorithme de complexité moindre dans notre cas (où le graphe est dense), qui est en  $O(n^2 \log n)$  mais qui nécessite toutefois une phase de tri des arêtes à chaque itération par tas de Fibonacci [16]. Nous n'avons pas éprouvé le besoin de programmer cette méthode car ce n'est pas l'appariement qu'il faut optimiser en priorité.

### 3.5. Récapitulatif.

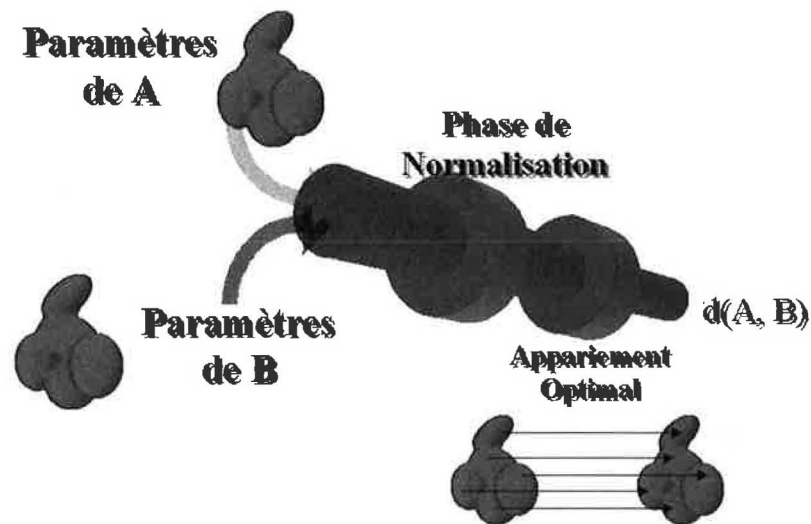


Figure 5

## 4. RÉSULTATS.

Nous présentons ici l'implantation et les applications du pipeline de mesure et développons en détail la méthode d'appariement optimal, cette dernière problématique restant bien sûr ouverte.

### 4.1. Comparaison d'objets.

La comparaison d'objets couvre le domaine de la confrontation d'objets à des fins d'analyse, et plus généralement les suivis temporels comme nous allons le voir. Nous proposons une approche multi-échelle qui enrichit les possibilités d'étude.

#### 4.1.1. La comparaison progressive.

La Figure 6 décrit la chaîne mise en œuvre pour calculer les indices de dissimilarité entre entités :

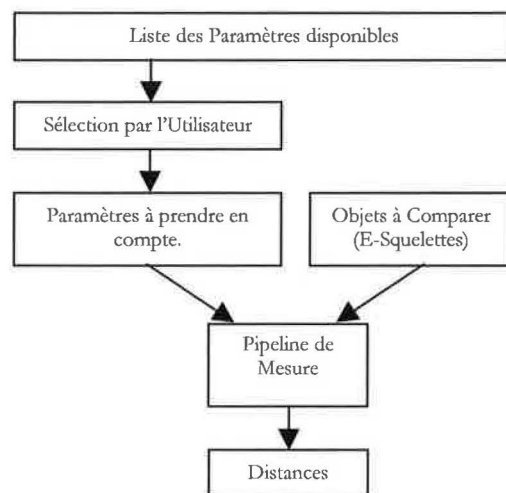


Figure 6

En sortie, la matrice de distance exhibe les valeurs d'indices de dissimilarité entre les E-Squelettes d'entrée suivant les paramètres pris en compte.

La Figure 7 montre la représentation par voxels de deux fémurs. Pour introduire une différence, nous avons ôté deux coupes scanner, ce qui représente une « cassure » de 14.6mm comme montré à gauche.

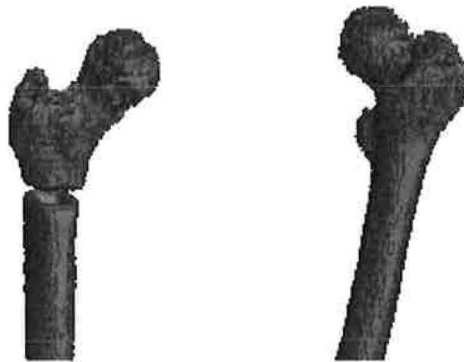


Figure 7

Nous réalisons alors la transformation en E-Squelette des deux objets. Observons l'évolution de l'indice de dissimilarité relativement au fémur de référence (attention, il n'est pas exprimé en pourcentage comme pourrait le laisser entendre la valeur maximale des ordonnées) :

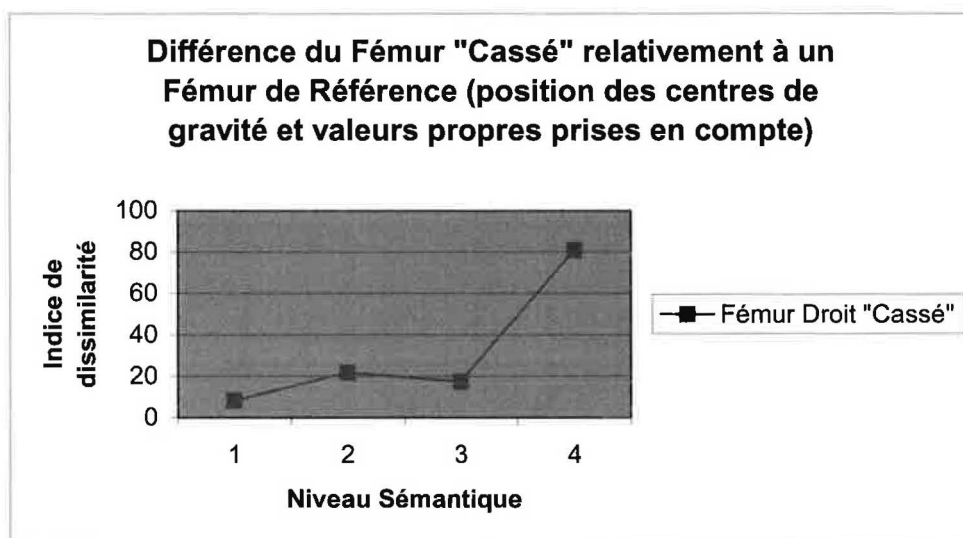


Tableau 1

Pour un nombre de classes égal à cinq, un pic apparaît dans le Tableau 1. Il est alors intéressant d'observer l'évolution de chacun des paramètres – à savoir la distance des centres de classe au centre de gravité local à l'objet et les amplitudes des vecteurs propres des classes – pour les niveaux sémantiques 3 et 4 :

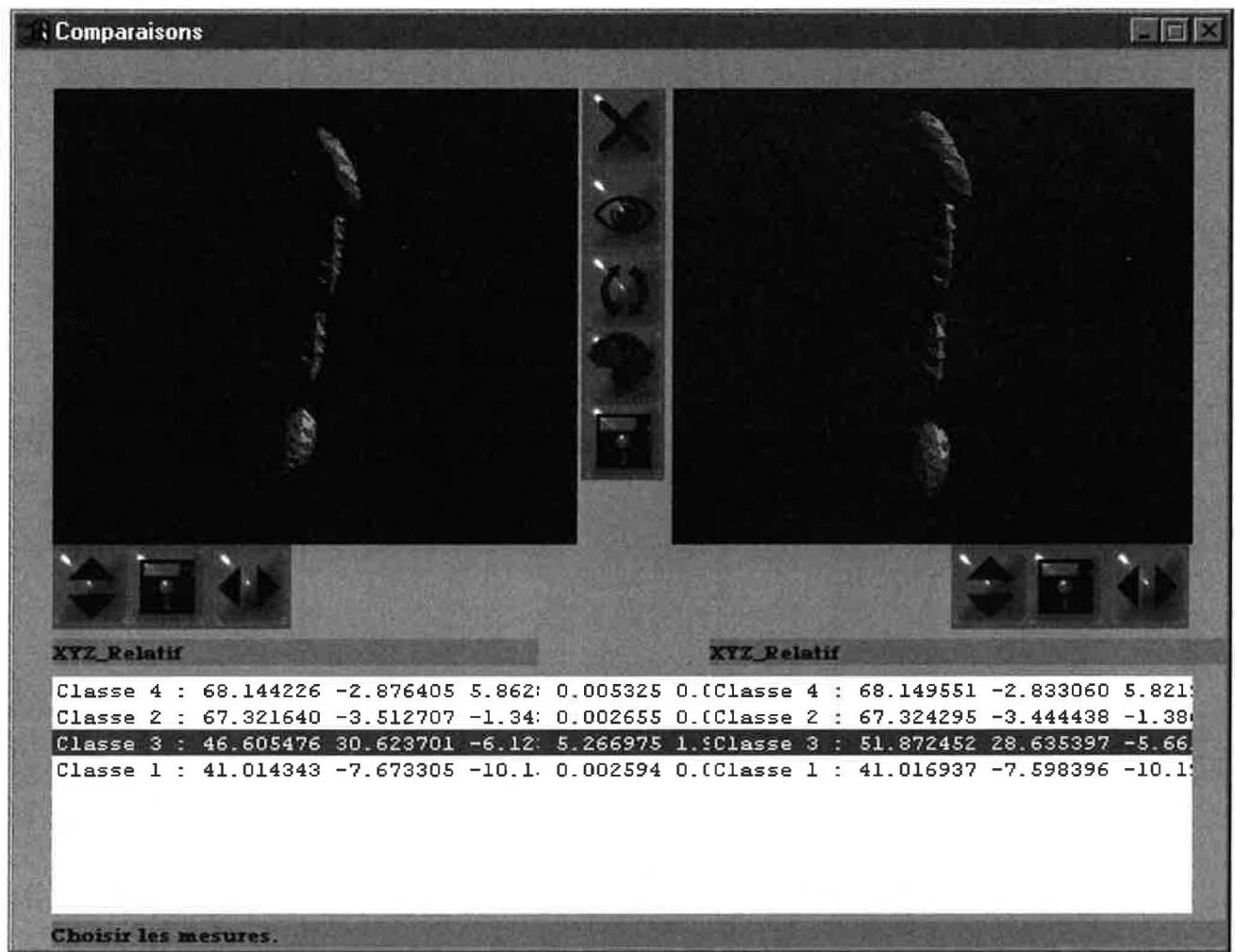


Figure 8

La Figure 8 montre une différence significative dans la classe numéro 3 (colorée en vert) pour le paramètre concernant l'amplitude des vecteurs propres (à gauche). Les points étant moins nombreux dans cette zone, il est normal que la classe soit « plus petite » que dans l'examen de référence (à droite). La détection d'anomalie est réalisée en tenant compte de la variation moyenne des classes relativement au fémur de référence : si celle-ci est largement supérieure par rapport à la moyenne des variations dans les autres classes, elle est alors mise en valeur. A ce stade, la position des centres de classes n'est pas modifiée de manière sensible.

Remarque :

Nous désignons par « amplitude des vecteurs propres » l'amplitude de la valeur propre associée à ce vecteur propre.

Le niveau sémantique 4 exhibe de nouvelles différences :



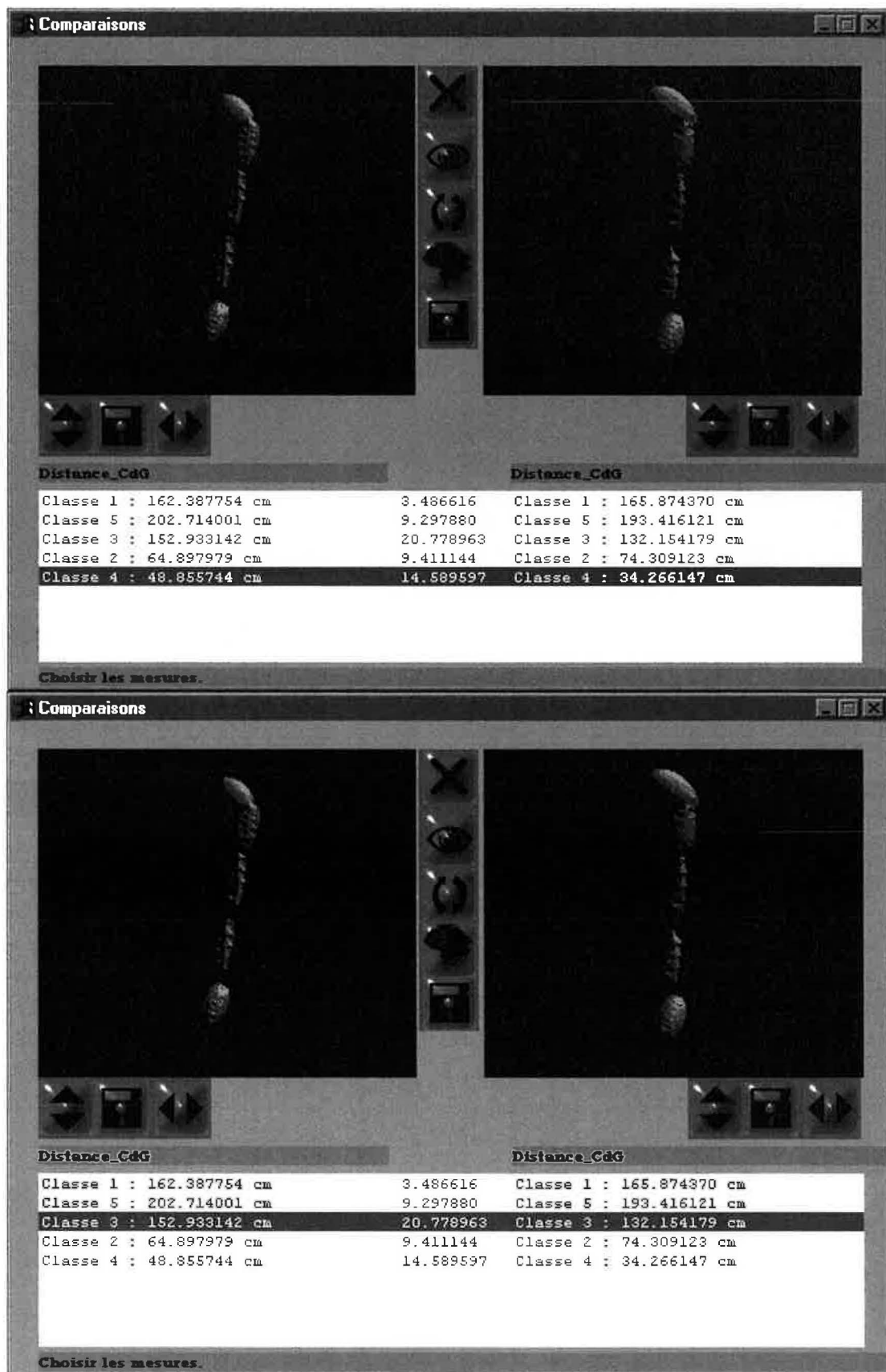


Figure 9

En effet, dans la Figure 9, la position des classes adjacentes à la « cassure » a été modifiée, respectivement de 14.6mm et 20.8mm. C'est le paramètre des positions des centres de classe qui est concerné.

Dans ce cas de figure, les classes adjacentes à la zone de différence ont pu être isolées. Par conséquent, nous disposons ici d'un moyen puissant de détection des variations locales entre deux E-Squelettes similaires. Cette technique est rendue possible par l'utilisation de la décomposition hiérarchique ou multi-échelle. C'est ce que nous appelons la *comparaison progressive*, qui révèle peu à peu les différences entre les objets à mesure que la distance de vision diminue.

#### 4.1.2. Le suivi.

En agissant sur les paramètres à injecter dans l'indice de similarité, l'éditeur d'analyse devient un outil puissant de recherche pour quantifier l'évolution d'une transformation structurelle. C'est le cas par exemple de la croissance osseuse.

Une étude a été menée conjointement avec le Docteur François Canovas, chirurgien orthopédique au Centre Hospitalier Universitaire Lapeyronie [2]. Elle met en évidence les changements dans la morphologie osseuse du carpe chez les enfants, ceci afin de détecter des pathologies ou des retards de croissance de manière précoce.

Le travail a été effectué sur un ensemble de 20 carpes d'enfants âgés de 4 à 14 ans. Les examens scanners ont été segmentés et chaque os a été identifié par le Dr. F. Canovas (cette identification ayant été réalisée avant la mise en place du module de reconnaissance automatique). Une fois cette opération effectuée, chaque os a été transformé en E-Squelette en intégrant les fonctions d'extraction de paramètres adéquates :

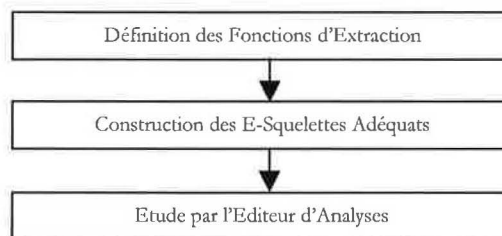


Figure 10

La Figure 10 présente le protocole suivi pour l'étude médicale. Cette méthodologie classique est généralisable à toute étude de phénomènes au travers des E-Squelettes.

L'éditeur d'analyse que nous avons développé autorise l'observation et la comparaison de E-Squelettes. La Figure 11 nous en présente l'interface :

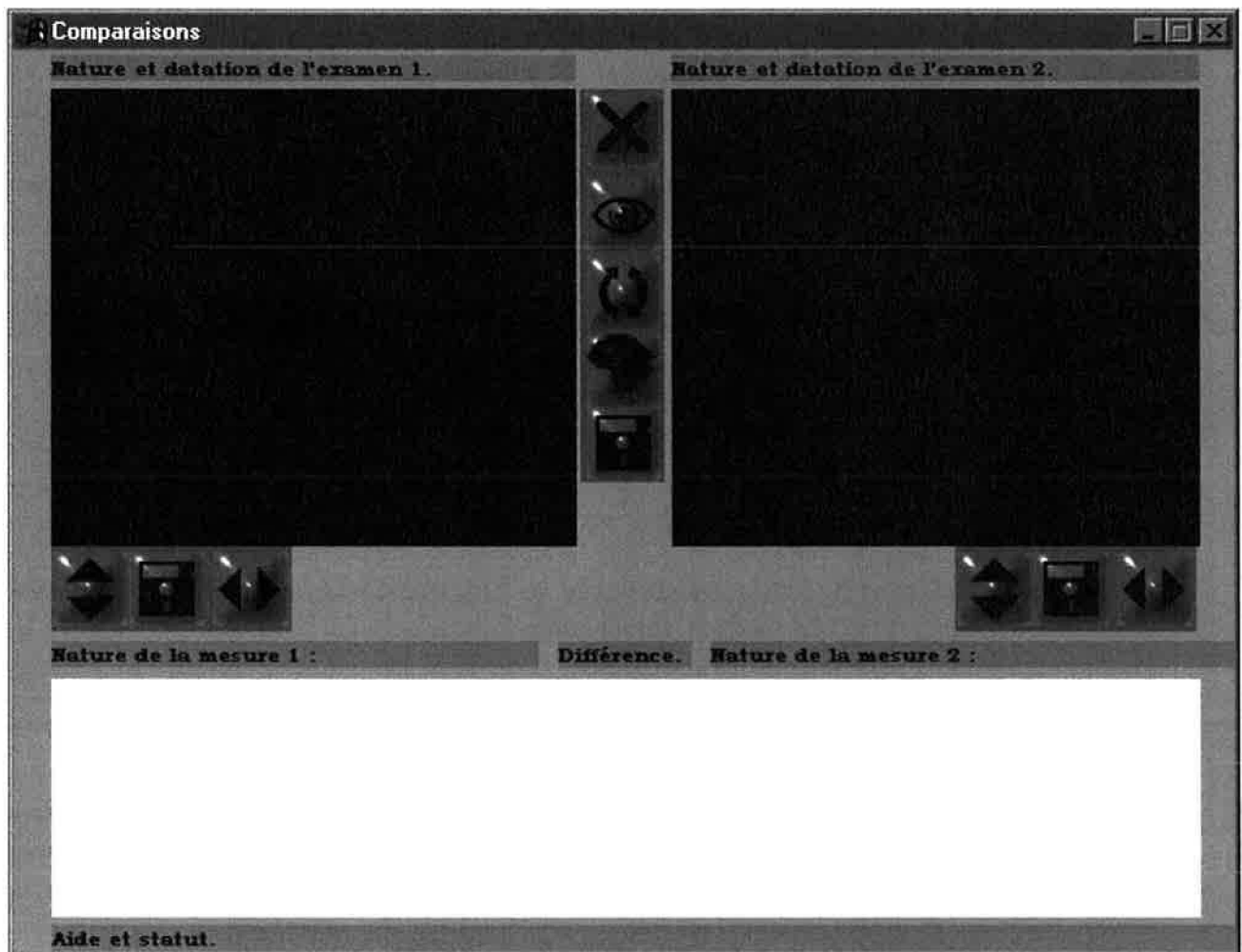


Figure 11

Une fois le E-Squelette chargé, il peut être visionné à différents niveaux sémantiques, et chaque paramètre peut être observé et comparé, la comparaison n'étant active que lorsque les sections gauche et droite sont au même niveau sémantique et que les mêmes paramètres sont sélectionnés.

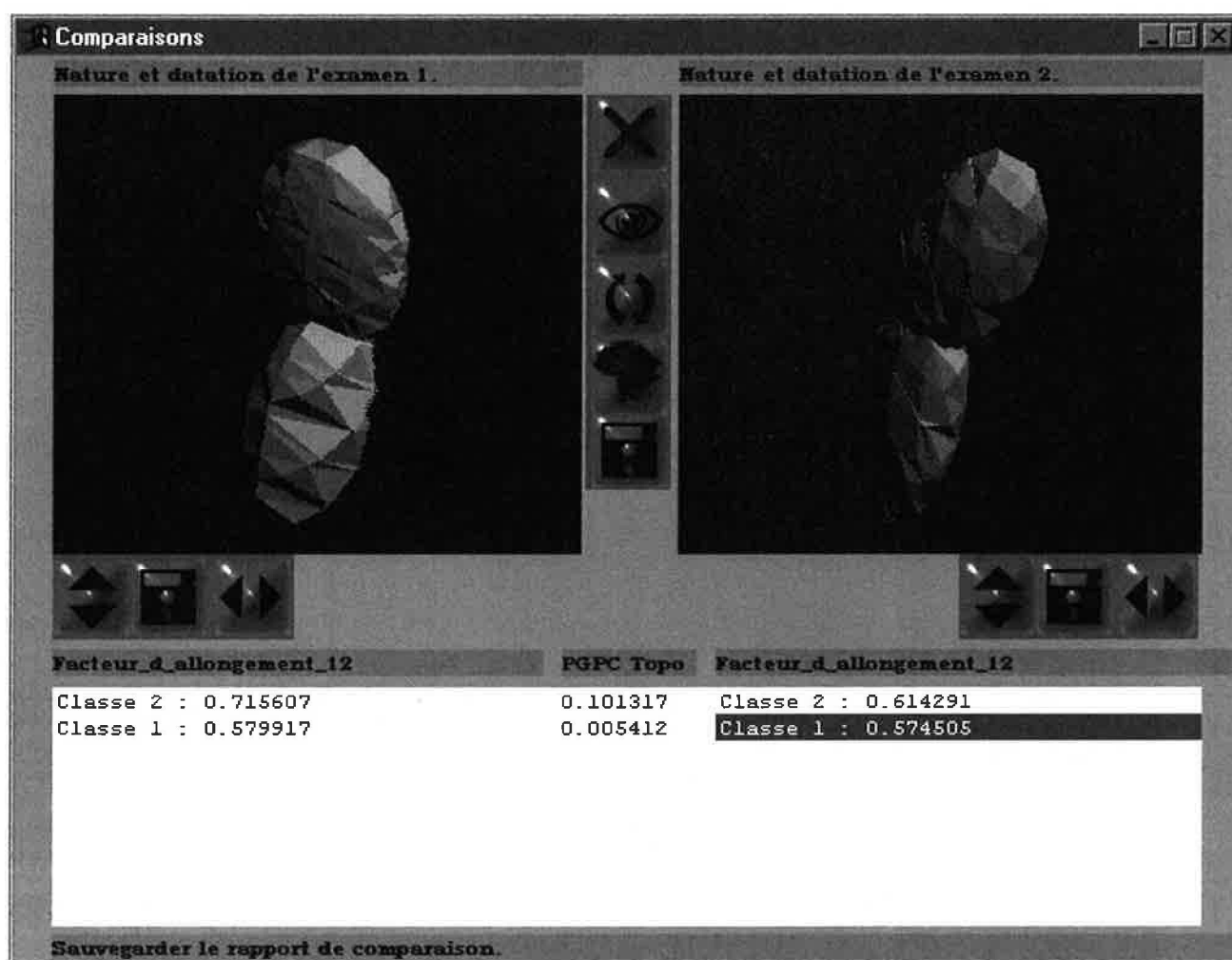


Figure 12

La Figure 12 montre deux E-Squelettes dont sont comparés les *facteurs d'allongement 1/2*, c'est-à-dire le rapport entre la longueur de l'axe secondaire d'inertie avec celle de l'axe principal d'inertie de chaque classe. La colonne centrale présente la valeur de la différence entre ces paramètres pour chaque classe. Dans la colonne de droite une classe est sélectionnée, ce qui se traduit par une coloration verte dans la vue 3D (colonne de droite).

L'étude menée par le Dr. Canovas vise à corrélér l'âge civil et l'âge osseux, et ce au travers de l'étude de la morphologie des os du carpe. Prenons l'exemple de l'*hamatum*, qui présente un particularité intéressante :

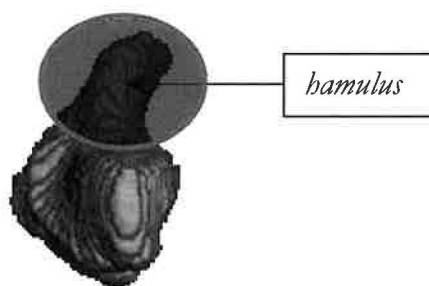


Figure 13

Comme indiqué dans la Figure 13, l'hamatum est doté d'une protubérance – dénommée *hamulus* – qui croît à mesure que la maturation osseuse se réalise. Il semble donc naturel de mesurer automatiquement et dans les trois dimensions la taille ainsi que la position de cette portion d'os :

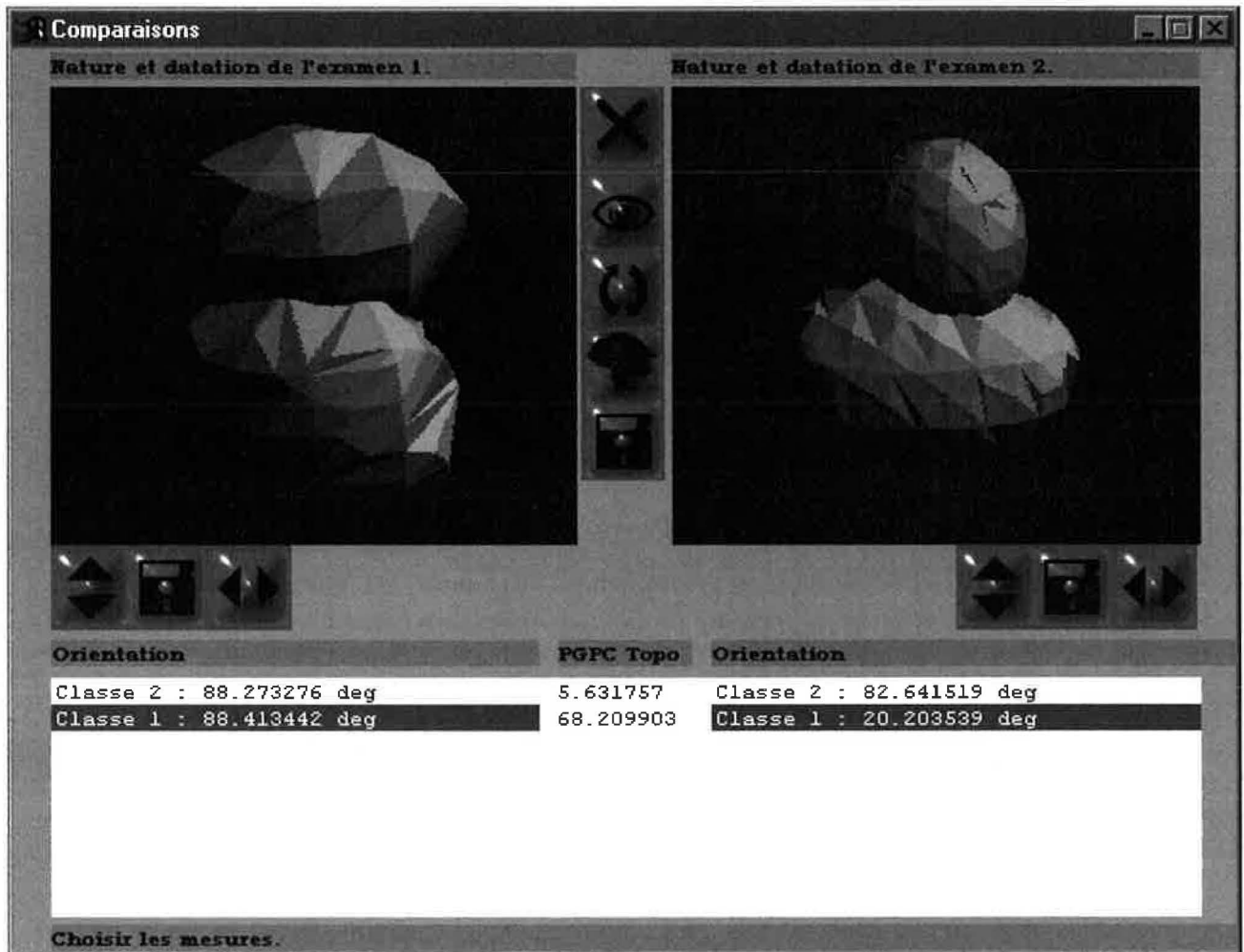


Figure 14

La Figure 14 montre à gauche le niveau sémantique 1 de l'hamatum d'un enfant de 4 ans, et à droite celui d'un enfant de 14 ans (hamulus vers le haut). Ce que l'on peut constater ici est l'absence de caractérisation de l'hamulus à gauche (les deux classes sont perpendiculaires à l'axe principal d'inertie de l'os entier). A droite, les deux parties sont clairement séparées : l'hamulus forme un angle de 20.2° avec l'axe principal d'inertie de l'hamatum, alors que la partie basse reste presque perpendiculaire à celui-ci.

Le lecteur voudra bien lire l'étude réalisée par le Dr. François Canovas [2] pour un développement plus complet, ainsi que ses travaux de thèse de doctorat en médecine [19]. Ce qui a été révélé grâce à l'éditeur d'analyse concerne les aspects suivants :

- il est difficile de trouver des facteurs de corrélation âge osseux/âge civil. La flexibilité du paramétrage et son intuitivité ont permis de tester rapidement plusieurs combinaisons de valeurs, en sélectionnant rapidement les bonnes caractérisations (exemple de l'hamulus) ;
- l'analyse quantitative dans ce cas de figure doit être ciblée selon certains critères. La difficulté pour trouver les corrélations provient de la variabilité morphologique liée essentiellement au sexe, mais aussi aux caractères génétiques propres à l'individu. L'observation de différents paramètres a mis en évidence ce phénomène. L'adjonction de fonctions de normalisation peut permettre de s'affranchir des facteurs homothétiques liés à la morphologie propre à chaque enfant ;
- des corrélations existent entre l'âge osseux et le rapport d'axes d'inertie pour un squelette de niveau sémantique 0. Au-delà, ce sont des transformations morphologiques (voir cas de l'hamatum), propres à chaque os, qui sont mises en évidence et qu'il reste encore à quantifier.

La flexibilité de l'outil autorise la recherche de relations à la fois inter-E-Squelettes et intra-E-Squelette. La détection des parties pertinentes au sens géométrique permet de dégager rapidement les structures et sous-structures qui composent le solide tridimensionnel.

#### **4.2. Reconnaissance d'objets.**

Les objets tridimensionnels **rigides** peuvent être reconnus. Si l'acquisition autorise une segmentation, l'ensemble des points ainsi détectés est un candidat potentiel à la transformation en E-Squelette. Une fois ces étapes franchies, il ne reste plus qu'à employer le pipeline de mesure couplé avec une bibliothèque de E-Squelettes d'objets connus, comme l'indique la Figure 15 :

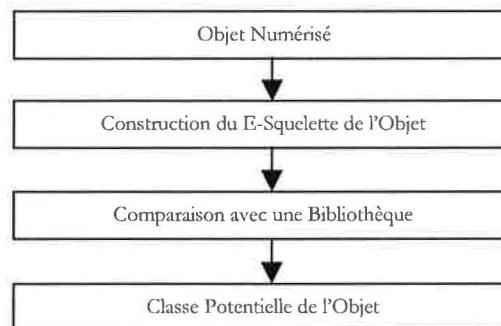


Figure 15

La structuration hiérarchique du E-Squelette autorise la *reconnaissance progressive par appariement de profils*. La technique que nous avons mise en place consiste donc à confronter les E-Squelettes de manière graduelle : les niveaux sémantiques sont d'abord

faibles, puis augmentent tant que l'indice de dissimilarité reste en deçà d'un seuil donné. Cette approche présente trois avantages principaux :

- accélération de la mesure : une grande différence entre deux objets sera détectée très tôt dans le processus, ce qui évitera un découpage inutile de l'objet. Au vu des complexités mises en jeu, cet aspect est intéressant ;
- aucune hypothèse n'est faite sur le choix du niveau sémantique puisqu'on fixe simplement un nombre maximum de classes pour lequel les deux objets seront considérés comme identiques ;
- la robustesse de la reconnaissance est augmentée puisqu'une moyenne d'indices est effectuée sur l'ensemble des mesures. C'est une manière de se préserver d'éventuelles sensibilités au bruit ou erreurs de détection.

Nous avons procédé à des tests de reconnaissance, que nous présentons maintenant :

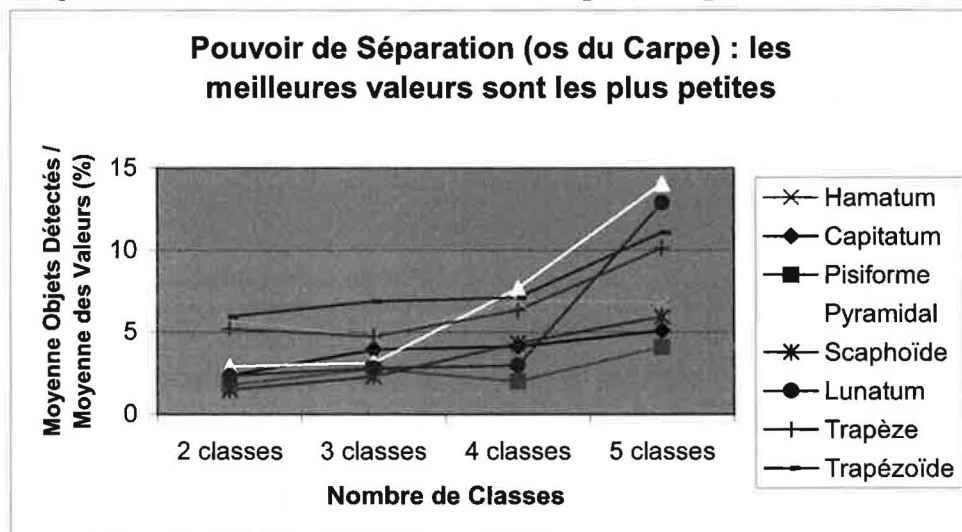


Tableau 2

Le Tableau 2 montre la qualité de la reconnaissance des os du carpe. Pour ce test, nous avons procédé comme suit :

1/ les 32 (huit os dans quatre examens) E-Squelettes sont successivement comparés à un examen de référence (huit os), et les indices de similarité sont stockés dans une matrice de distance. Les données intégrées dans la mesure de l'indice de similarité sont la *distance des centre de gravité des classes par rapport au centre de gravité local* de l'objet et *l'amplitude des vecteurs propres*.

2/ Dans chaque ligne, sont choisies les quatre plus petites valeurs non nulles puisqu'il y a quatre examens confrontés à l'examen de référence.

3/ Le rapport suivant, nommé *pouvoir de séparation* (converti en pourcentage) est alors évalué :



$$100. \frac{\text{Moyenne des quatre indices de similarité les plus petits}}{\text{Moyenne de tous les indices de similarité}}$$

Cet indice permet d'évaluer la différenciation des objets relativement à la moyenne des distances par rapports à tous les autres : un faible pourcentage indiquera que les objets identiques à l'objet de référence sont beaucoup plus proches de ce dernier que tous les autres.

Bien entendu il a été vérifié que les quatre indices de similarité les plus petits correspondaient bien aux objets de même nature que l'objet de référence. Dans le cas du carpe, aucune erreur n'a donc été commise, même si le pouvoir de séparation décroît lorsque le niveau sémantique croît, ce qui était prévisible (les différences apparaissent progressivement et chaque objet exhibe ses différences, essentiellement à cause de variations dans le processus de segmentation dans les zones périphériques : l'incertitude de la densité sur les bords de l'objet est importante).

Un extrait de la matrice des distances pour trois classes est donné dans le tableau suivant :

os	H1	H2	H3	H4	H5	C1	C2	C3	C4	C5	L1	L2	L3	L4	L5
H1	0	8	3	3	3	67	65	64	63	65	54	53	52	53	53
H2	8	0	8	7	7	69	67	66	66	67	54	53	52	54	53
H3	3	8	0	4	4	67	65	64	63	65	53	52	52	52	52
H4	3	7	4	0	1	68	66	65	64	66	53	53	51	53	53
H5	3	7	4	1	0	68	66	65	65	66	53	53	52	53	53
C1	67	69	67	68	68	0	4	5	6	3	68	66	68	68	66
C2	65	67	65	66	66	4	0	6	7	3	66	64	65	65	64
C3	64	66	64	65	65	5	6	0	3	5	68	66	68	67	66
C4	63	66	63	64	65	6	7	3	0	6	67	65	66	66	65
C5	65	67	65	66	66	3	3	5	6	0	67	65	66	66	65
L1	54	54	53	53	53	68	66	68	67	67	0	5	2	3	4
L2	53	53	52	53	53	66	64	66	65	65	5	0	4	3	1
L3	52	52	52	51	52	68	65	68	66	66	2	4	0	2	3
L4	53	54	52	53	53	68	65	67	66	66	3	3	2	0	2
L5	53	53	52	53	53	66	64	66	65	65	4	1	3	2	0

Tableau 3

Le Tableau 3 indique la matrice de distance concernant l'Hamatum (H1...5 dans le Tableau 3), le Capitatum (C1...5) et le Lunatum (L1...5). Les valeurs ont été regroupées et arrondies à l'entier supérieur pour mettre en évidence la qualité de séparation.

Nous avons également utilisé un autre examen ; il s'agit d'une acquisition par scanner des mâchoires inférieure et supérieure d'un adulte. Les dents ont été segmentées et identifiées par un expert. Ses caractéristiques sont indiquées dans le Tableau 4 :

Examen	Pixels en X	Pixels en Y	Taille de Pixel	Hauteur (Z)	#coupes
Dents	407	412	0.49mm	1.5mm	102



Tableau 4

Les comparaisons sont possibles entre deux héli-arcades, la symétrie s'opérant relativement au plan *sagital* (profil).

	11	21	31	41	15	25	35	45
11	0.	20.1	27.4	27.1	34.9	33.3	26.5	29.4
21	20.1	0.	29.5	30.3	38.6	35.4	24.4	27.6
31	27.4	29.5	0.	3.7	18.9	19.4	28.7	26.
41	27.1	30.3	3.7	0.	18.6	18.7	28.2	26.7
15	34.9	38.6	18.9	18.6	0.	4.4	41.6	40.4
25	33.3	35.4	19.4	18.7	4.4	0.	39.3	36.9
35	26.5	24.4	28.7	28.2	41.6	39.3	0.	8.9
45	29.4	27.6	26.	26.7	40.4	36.9	8.9	0.

Tableau 5

Dans le Tableau 5 sont comparées les dents des deux héli-arcades supérieures, à savoir les incisives, canines et prémolaires, et ce sans tenir compte de la symétrie (pas de distinction gauche/droite). Les zones grisées indiquent la mise en correspondance des dents de même nature. La valeur 20.1 pour les dents **11** et **21** met en évidence une pathologie sur l'incisive : l'indice reste minimal, mais la qualité de séparation est insuffisante. Il est possible par cette méthode de détecter des problèmes inhérents à des objets déjà connus, comme nous l'avions mis en évidence dans la section 4.1.

## 5. CONCLUSION.

La détermination des paramètres les plus pertinents pour représenter ou *caractériser* au mieux les objets est cruciale [6]. C'est ce choix qui influence le plus les résultats de classification. Le nombre de paramètres par exemple peut être trop grand ce qui, corrélé avec un ensemble de tests trop faible, entraîne des erreurs de reconnaissance.

De manière générale, il faut garder à l'esprit les points suivants :

- tout algorithme de partitionnement aboutit à un résultat, qu'il soit bon ou mauvais ;
- il n'existe pas de « meilleur » algorithme de partitionnement.

Nous avons porté notre réflexion sur un paramétrage intuitif des techniques mises en œuvre pour le E-Squelette. Le découpage par les Nuées Dynamiques reste par exemple très prévisible, tout en donnant des résultats satisfaisants pour la caractérisation géométrique. Une telle technique est adaptée à la puissance de calcul actuelle malgré une complexité non linéaire dans la taille de la donnée. Se baser sur la géométrie du solide est également une approche très concrète, avec de surcroît la possibilité d'enrichir la représentation en extrayant des valeurs supplémentaires comme la densité moyenne, le kurtosis...Etc.

Ces choix constituent pour nous une tentative de contournement des problèmes de réduction de dimension et de minimisation d'erreur de classification. En effet, même si la mesure de qualité est adaptée – ce qui reste complexe à mettre œuvre - , il reste à évaluer toutes les combinaisons de paramètres possibles pour sélectionner la meilleure d'entre elles. Il a été montré qu'aucune méthode séquentielle non exhaustive (test de toutes les possibilités) ne garantissait la production de la solution optimale [4]. La méthode du *branch and bound* est une solution, mais elle repose sur la propriété de monotonie du critère d'optimisation, ce qui n'est pas toujours le cas. Enfin, les autres techniques apportent une solution mais ne garantissent plus l'optimalité [5].

Il est notoire que le processus de reconnaissance humaine repose essentiellement sur le contexte et l'expérience. De cette observation découle notre orientation vers une meilleure appréhension de la technique par l'expert qui choisira les meilleurs paramètres sans être handicapé par la complexité du moteur de reconnaissance.

## 6. BIBLIOGRAPHIE.

- [1] F. Banégas, Dr. F. Canovas, D. Michelucci, M. Roelens et M. Jaeger, « *Hierarchical Pattern Analysis and Recognition Using Ellipsoidal Skeleton.* », 2000, Fourth International Conference in Computer Graphics and Artificial Intelligence, 3IA'2000, Limoges, France.
- [2] F. Canovas, F. Banégas, C. Cyteval, M. Jaeger, A. Dimeglio, C. Sultan, F. Bonnel, « *Carpal Bone Maturation Assessment by Image Analysis from CT-scans.* », International Journal of Experimental and Clinical Endocrinology, Hormone Research, Karger Eds, 2000.
- [3] D. de Werra, « *Elements de Programmation Linéaire avec Application aux Graphes.* », Presses Polytechniques Romandes, 1990.
- [4] A. K. Jain, R. P. W. Duin and J. Mao, « *Statistical Pattern Recognition : A Review.* », IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), pp. 4-34, January, 2000.
- [5] T. M. Cover, J. M. Van Campenhout, « *On the Possible Orderings in the Measurement Selection Problem.* », IEEE Transactions on Systems, Man and Cybernetics, 7(9), pp. 657-661, September, 1977.
- [6] A. K. Jain, D. Zongker, « *Feature Selection : Evaluation, Application and Small Sample Performance.* », IEEE Transactions on PAMI, 19(2), pp. 153-158, February, 1997.
- [7] D. Shen, H. S. Horace, K. T. Cheung and E. A. Teoh, « *Symmetry Detection by Generalized Complex (GC) Moments : A Close-Form Solution.* », IEEE Transactions on PAMI, 21(5), May, 1999.
- [8] H. Hügli, C. Schütz, « *Geometric Matching of 3D Objects : Assessing the Range of Successful Configurations.* », International Conference of Recent Advances in 3D Digital Imaging and Modeling, May 12-15, Ottawa, Ontario, Canada, 1997.
- [9] K. Zhang, « *A Constrained Edit Distance Between Unordered Labeled Trees.* », Algorithmica, 15, pp. 205-222, Springer-Verlag New York Inc., 1996.
- [10] J. Martin, A. Pentland, S. Sclaroff and Ron Kikinis, « *Characterization of Neuropathological Shape Deformations.* », IEEE Transactions on PAMI, 20(2), pp. 97-112, February, 1998.
- [11] E. P. Simoncelli, « *A Rotation-Invariant Pattern Signature.* », Proceedings of 3<sup>rd</sup> IEEE International Conference on Image Processing, III, pp. 185-188, IEEE Signal Processing Society, Lausanne, Switzerland, 16-19 september, 1996.
- [12] L. R. Ford Jr., D. R. Fulkerson, « *Maximal Flow through a Network.* », Canadian Journal of Mathematics, 8, pp. 399-404, 1956.
- [13] L. R. Ford Jr., D. R. Fulkerson, « *Flow in Networks.* », Princeton University Press, Princeton, NJ, 1962.
- [14] J. Edmonds, R. M. Karp. « *Theoretical improvements in algorithmic efficiency for network flow problems.* », Journal of the ACM, 19(2):248-264, April 1972.
- [15] Z. Galil, « *Efficient Algorithms for Finding Maximum Matching in Graphs.* », Computing Surveys, vol. 18, No. 1, March 1986.
- [16] Z. Galil, S. Micali and H. Gabow, « *An  $O(EV \log V)$  Algorithm for Finding a Maximal Weighted Matching in General Graphs.* », SIAM J. COMPUT., vol. 15, No. 1, February 1986.

- [17] D. de Werra, « *Eléments de Programmation Linéaire avec Application aux Graphes.* », Presses Polytechniques Romandes, p. 200, 1990.
- [18] R. Tam, A. Fournier, « *Image Interpolation Using Union of Spheres.* », The Visual Computer, 14, 8/9, pp. 401-414, 1998.
- [19] F. Canovas, « *Etude de la Maturation des Os du Carpe par Analyse d'Image à partir de Coupes Tomodensitométriques du Poignet.* », Mémoire de thèse, Université Montpellier I, U.F.R. de Médecine, Formation doctorale : Endocrinologie cellulaire et moléculaire, Discipline : Anatomie, 24 octobre 2000.



## ∞. Conclusion.

Nous avons proposé un modèle de caractérisation de solides tridimensionnels, offrant des capacités de visualisation, de reconstruction surfacique fine et de reconnaissance automatique. Chaque domaine apporte son lot de nouvelles voies, et chaque voie explorée apporte son lot de difficultés – ce qui laisse entrevoir ce qui reste encore à faire...

En restant pragmatiques, nous avons cernés les quelques thèmes cruciaux qui apporteraient de grandes possibilités au E-Squelette :

- Représentation tridimensionnelle :
  - augmentation du répertoire de primitives ;
  - amélioration de l'heuristique liée à la méthode tabou, pour la perturbation des paramètres notamment ;
  - regroupement de primitives, par exemple deux cylindres adjacents peuvent être fusionnés en un seul sous certaines conditions...Etc. ; de cette manière il serait possible de reconstruire les structures tubulaires sans nécessité d'une multitude de primitives ;
  - reconstruction de concavités ou d'objets creux par soustraction implicite, le plus difficile étant de détecter et de représenter ces « creux » dans le formalisme du E-Squelette.
- Caractérisation des solides :
  - étudier la mise en place de distances « intuitives » entre objets, selon des critères adaptés ;
  - explorer les possibilités de partitionnement, en spécialisant les paramètres de la chaîne de construction du E-Squelette et trouver ainsi des décompositions plus pertinentes selon les domaines concernés ;
  - détecter des hallométries dans des organes rigides à partir de la décomposition en E-Squelette ;
  - générer des organes synthétiques à partir d'hallométries connues sur des organes rigides.

Ces points essentiellement techniques ne décrivent qu'une partie de l'évolution possible du E-Squelette. Nous pensons avec conviction qu'il n'existe pas de « E-Squelette générique », chaque domaine possède ses propres caractéristiques, comme chaque événement à son symbole. C'est le cadre que nous désignons comme générique, en ce sens qu'il peut supporter une large gamme de méthodes différentes, ce qui le transforme alors plus en application qu'en outil intrinsèquement expérimental. Nous avons à l'esprit que c'est l'expert qui, *in fine* :

- décidera de ce qui est pertinent ;
- définira ce qu'il appelle « paramètre intuitif » ;
- appréciera la qualité du résultat de la méthode automatique ;

- supervisera les résultats de cette même méthode.

Cette réalité est on ne peut plus vraie en médecine, où les contacts se nouent avec des personnes qui ne connaissent que peu l'informatique et ne sont pas forcément conscients de ses limitations ou *a contrario* de ses possibilités. Le cadre formel du E-Squelette est une tentative de réponse à la question : comment fournir un outil de visualisation et de caractérisation que l'expert externe à l'informatique peut maîtriser et faire évoluer rapidement ? La partie « évolution » requiert bien sûr un informaticien puisqu'elle demande de la programmation, mais le cadre fournit un support d'intuition pour l'expert qui désire tel ou tel comportement ou raffinement de manière précise dans la chaîne de construction. Le dialogue étant toujours difficile lorsque deux spécialités totalement étrangères l'une de l'autre se rencontrent, le besoin d'une base commune est pressant. Cet aspect d'interface reste donc largement à développer, et c'est sans nul doute ce qui fera évoluer le plus le E-Squelette.





# **Annexe 1. Publications.**

Sont présentées dans cette section les publications réalisées durant ce travail de thèse, classées chronologiquement. Les références officielles sont les suivantes :

Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Partitionnement Hiérarchique d'un Nuage de Points par Squelette Ellipsoïdal.**', 1998, Proceedings of 6ièmes Journées de l'Association Française d'Informatique Graphique, Dunkerque, France.

Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Application of Ellipsoidal Skeleton to Organ Geometric Modeling from CT-scan Images.**', Poster Session 3661-76, 1999, Proceedings of the SPIE Conference on Medical Imaging '99, San Diego, California. (conférence à comité de lecture)

Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Automatic Extraction of Significant Features from 3D Point Clouds by Ellipsoidal Skeleton. Applications in Vision and Geometrical Characterization.**', 1999, Proceedings of the International Conference in Visual Computing '99, pp. 58-67, Goa, India. (conférence à comité de lecture)

Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Ellipsoidal Skeleton for Multi-Scaled Solid Reconstruction.**', 1999, Proceedings of the Swiss Conference of CAD/CAM'99, pp. 33-40, Neuchâtel, Switzerland. (conférence à comité de lecture)

Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Automatic Adaptive Surface Reconstruction from Ellipsoidal Skeleton.**', 1999, Proceedings of The Fourth International Workshop on Implicit Surfaces '99, ACM Press, in Bordeaux, France. (conférence à comité de lecture)

Frédéric Banégas, Stéphane Chemouny, '**Une Chaîne Complète d'Analyse en Imagerie Médicale.**', 1999, Poster présenté à la Semaine de la Science en Fête '99, Montpellier, France.

Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, '**Hierarchical Pattern Analysis and Recognition Using Ellipsoidal Skeleton.**', 2000, Proceedings of the Fourth International Conference in Computer Graphics and Artificial Intelligence, 3IA'2000, Limoges, France. (conférence à comité de lecture)

F. Canovas, F. Banégas, C. Cyteval, M. Jaeger, A. Dimeglio, C. Sultan, F. Bonnel, '**Carpal Bone Maturation Assessment by Image Analysis from CT-scans.**', published in Radiology. (revue médicale à comité de lecture)

# **Annexe 2. Bibliographie Globale.**

Nous proposons ici une fusion des bibliographies présentées dans ce mémoire, avec le même numéro que dans le chapitre correspondant.

## **1. LA RÉALITÉ VIRTUELLE.**

- [1] G. Burdea, P. Coiffet, « La Réalité Virtuelle », Hermès, Paris, 1993
- [2] G. Francastel, « L'Image, La Vision et L'Imagination : De la Peinture Au Cinéma », Denöel/Gonthier, 1983
- [3] M. Göbel, « Projects in VR », IEEE Computer Graphics and Applications, pp. 10-13, Volume 16, Number 1, January 1996
- [4] P. Dev, « Imaging and Visualization in Medical Education », IEEE Computer Graphics and Applications, pp.21-31, Volume 19, Number 3, May/June 1999
- [5] J. Charles, « Virtually There : Taking VR to Work », IEEE Computer Graphics and Applications, pp.4-9, Volume 17, Number 6, November/December 1997
- [6] R. Baldwin, « VR : Friend or Foe ? », IEEE Computer Graphics and Applications, pp.102-103, Volume 17, Number 6, November/December 1997
- [7] Y. Ikei, K. Wakamatsu and S. Fukuda, « Vibratory Tactile Display of Image-Based Textures », IEEE Computer Graphics and Applications, pp.53-61, Volume 17, Number 6, November/December 1997
- [8] S. L. Dawson, J. A. Kaufman, « The Imperative for Medical Simulation », IEEE Special Issue on Virtual & Augmented Reality in Medicine, pp. 479-483, Volume 86, Numero 3, March 1998
- [9] R. M. Satava, S. B. Jones, « Current and Future Applications of Virtual Reality for Medicine », pp. 484-489, Volume 86, Numero 3, March 1998
- [10] E. Chen, B. Marcus, « Force Feedback for Surgical Simulation », pp. 524-529, Volume 86, Numero 3, March 1998
- [11] Z. Soferman, D. Blythe, « Advanced Graphics Behind Medical Virtual Reality : Evolution of Algorithms, Hardware, and Software Interfaces », pp. 531-554, Volume 86, Numero 3, March 1998
- [12] J. W. Hill, « Telepresence Technology in Medicine : Principles and Applications », pp. 569-580, Volume 86, Numero 3, March 1998
- [13] N. Ezquerro, I. Navazo, T. I. Morris and E. Monclus, « Graphics, Vision, and Visualization in Medical Imaging : A State of the Art Report », pp.21-80, Eurographics'99 STAR, 20<sup>th</sup> Annual Conference of The European Association for Computer Graphics « Bringing to a New Life our Cultural Heritage », 7—11 September 1999, Milano
- [14] H. Iwata, « The Torus Treadmill : Realizing Locomotion in Ves », IEEE Computer Graphics and Applications, pp. 30-35, November/December 1999
- [15] H. Hoffman, J. Groen, S. Rousseau, A. Hollander, W. Winn, M. Wells and T. Furness, « Tactile Augmentation : Enhancing Presence in Virtual Reality with Tactile Feedback from Real Objects" » Journal of American Psychological Society, 1996, San Francisco, CA
- [16] S. Rosenberg, « The New Video-Game Philisophers : Did Someone Say Suspension of Disbelief ? », Digital Culture, May 17, 1995
- [17] F. P. Brooks, Jr., « What's Real About Virtual Reality ? », IEEE Computer Graphics and Applications, pp. 16-27, November/December 1999
- [18] E. Sutherland, « The Ultimate Display », Information Processing 1965 : Proceedings of IFIP Congress 65, Volume 2

- [19] E. M. Kolasinski, S. L. Goldberg, J. H. Hiller, « Simulator Sickness in Virtual Environments », U.S. Army Research Institute for the Behavioral and Social Sciences, Technical Report 1027, May 1995
- [20] G. Burdea, « Virtual Reality Systems and Applications », Electro'93 International Conference, Short Course, Edison NJ, 28<sup>th</sup> April, 1993
- [21] F. Brooks *et al.*, « Project GROPE – Haptic Displays for Scientific Visualization », Computer Graphics, Volume 24, Number 4, pp. 177-185
- [22] B-T Phong, « Illumination for Computer Generated Pictures », CACM, 18(6), pp. 311-317, June 1975
- [23] H. Gouraud, « Continuous Shading of Curved Surfaces », IEEE Transactions on Computers, C-20(6), pp.623-629, June 1971
- [24] W. Gibson, « Neuromancien », Edition J'ai Lu
- [25] J. G. Casali, « Vehicular Simulation-Induced Sickness, Volume 1 : An Overview. », IEOR Technical Report 8501 (Naval Training System Center TR 86-010), August 1986
- [26] R. S. Kennedy, J. E. Fowlkes, « Simulator Sickness is Polygenic and Polysymptomatic : Implications for Research », International Journal of Aviation Psychology, 2(1), pp. 23-38, 1992
- [27] R. S. Kennedy, J. E. Fowlkes, K. S. Berbaum and M. G. Lilienthal, « Use of a Motion Sickness History Questionnaire for Prediction of Simulator Sickness », Aviation, Space and Environmental Medicine, 63(7), pp. 588-593, 1992
- [28] M. E. Mc Cauley, T. J. Sharkey, « Cybersickness : Perception of Self-Motion in Virtual Environments », Presence, 1(3), pp. 311-318, 1992
- [29] M. Treisman, « Motion Sickness : An Evolutionary Hypothesis », Science, 197, pp. 493-495, 1977
- [30] T. A. Stoffregen, G. E. Riccio, « An Ecological Theory of Orientation and the Vestibular System" » Psychological Review, 95(1), pp. 3-14, 1988
- [31] T. A. Stoffregen, G. E. Riccio, « An Ecological Critique of the Sensory Conflict Theory of Motion Sickness" » Ecological Psychology, 3(3), pp. 15-194, 1991
- [32] J. T. Reason, J. J. Brand, « Motion Sickness », London : Academic Press, 1975
- [33] E. C Regan, « Side-Effects of Immersive Virtual Reality », International Applied Military Psychology Symposium, July 26-29, 1993
- [34] R. M. Stern, S. Hu, R. Leblanc and K. L. Koch, « Chinese Hyper-Susceptibility to Vection-Induced Motion Sickness », Aviation, Space and Environmental Medicine, (9), pp. 827-830, 1993
- [35] R. S. Kennedy, K. S. Berbaum, M. G. Lilienthal, W.P. Dunlap, B. F. Mulligan and J. F. Funaro, « Guidelines for Alleviation of Simulator Sickness Symptomatology », (NTSC TR 87-007), March, 1987
- [36] K. C. Uliano, E. Y. Lambert, R. S. Kennedy and D. J. Sheppard, « The Effects of Asynchronous Visual Delays on Simulator Flight Performance and the Development of Simulator Sickness Symptomatology », (NTSC 85-D-0026-1), 1986
- [37] R. S. Kennedy, L. H. Franck, « A Review of Motion Sickness with Special Reference to Simulator Sickness », Workshop on Simulator Sickness, 26-28 September, 1983

- [38] D. E. Parker, D. L. Harm, « Mental Rotation : A Key to Mitigation of Motion Sickness in the Virtual Environment », *Presence*, 1(3), pp. 329-333, 1992
- [39] J. Ehrlich, M. J. Singer and S. Cinq-Mars, « Stereoscopic and Monoscopic Displays in Virtual Environments », ARI Technical Report, Research Institute for the Behavioral and Social Sciences, Alexandria, VA
- [40] K. R. Boff, J. E. Lincoln, « Engineering Data Compendium : Human Perception and Performance », AAMRL, Wright-Patterson AFB, OH, pp. 166-191, 1988
- [41] T. J. Sharkey, M. E. McCauley, « Does a Motion Base Prevent Simulator Sickness ? », *Proceedings of the AIAA/AHS Flight Simulation Technologies Conference*, Report AIAA-92-4133-CP, Washington D.C., pp. 21-28, 1992
- [42] R. M. Satava, « Virtual Reality Surgical Simulator – The First Steps », *Proceedings of VR Systems '93 Conference*, New York, pp. 41-49, March, 1993
- [43] Y. Ikei, K. Wakamatsu and S. Fukuda, « Vibratory Tactile Display of Image-Based Textures », *IEEE Computer Graphics and Applications*, pp. 53-61, Volume 17, Number 6, November/December 1997
- [44] Asimov, « *Fantastic Voyage* », Boston, MA, Houghton-Mifflin, 1966
- [45] W. M. Smith, « Scanning the Technology : Engineering and Medical Science Chart *Fantastic Voyage* », *IEEE Special Issue on Virtual and Augmented Reality in Medicine*, pp.474-478, Volume 86, Number 3, March 1998



## **2. TOUR D'HORIZON DE LA REPRÉSENTATION TRIDIMENSIONNELLE.**

- [1] F. Banégas, « Zoom Sémantique par Maillage Progressif », rapport de stage de DEA, CIRAD/ENSMSE, 1997
- [2] P. J. Frey, H. Borouchaki, « Texel : Triangulation de Surfaces Implicites. Partie I : Aspects Théoriques », RR N°3066, INRIA, 13 dec 1996
- [3] P. J. Frey, H. Borouchaki, « Texel : Triangulation de Surfaces Implicites. Partie II : Exemple d'Applications », RR N°3066, INRIA, 13 dec 1996
- [4] J. Wilhelms, A. V. Gelder, « A Coherent Projection Approach for Direct Volume Rendering », pp. 275-284, Computer Graphics, Proceedings of SIGGRAPH'91, Volume 25, Number 4, July 1991
- [5] H.K. Tuy, L. T. Tuy, « Direct 2D Display of 3D Objects », pp. 29-33, IEEE Computer Graphics and Application, Volume 4, Number 10, October 1984
- [6] C. Upson, M. Keeler, « The V-Buffer : Visible Volume Rendering », pp. 59-64, Computer Graphics, Proceedings of SIGGRAPH'90, Volume 22, Number 4, July 1990
- [7] M. Levoy, « Volume Rendering, A Hybrid Ray Tracer for Rendering Polygon and Volume Data », pp. 33-40, IEEE Computer Graphics and Applications, Volume 10, Number 2, March 1990
- [8] P. Sabella, « A Rendering Algorithm for Visualizing 3D Scalar Fields », pp. 51-58, Computer Graphics, Volume 22, Proceedings of SIGGRAPH'88, Number 4, July 1988
- [9] P. Hanrahan, D. Laur, « Hierarchical Splatting : A Progressive Refinement Algorithm for Volume Rendering », pp. 285-288, Computer Graphics, Volume 25, Number 4, Proceedings of SIGGRAPH'91, August 1991
- [10] L. Westover, « Footprint Evaluation for Volume Rendering », pp. 367-376, Computer Graphics, Volume 24, Number 4, Proceedings of SIGGRAPH'90, August 1990
- [11] J. P. Thirion, A. Gourdon, « Computing the Differential Characteristics of Isodensity Surfaces », pp. 190-202, CVGIP, 1995.
- [12] W. J. Schroeder, J.A. Zarge and W. Lorensen, « Decimation of Triangle Mesh », pp. 65-70, ACM Computer Graphics, Proceedings of SIGGRAPH'92, Volume 26, Number 2, 1992
- [13] M. S. Sheppard, M. K. Georges, « Automatic Three Dimensional Mesh Generation by the Finite Octree Technique », pp.709-749, Int. J. Numer. Mthods Eng., Volume 32, 1991
- [14] H. Hoppe, « Progressive Meshes », pp. 99-108, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996
- [15] P. Lindstrom, D. Koller, W. Ribarsky, L. F. Hodges, N. Faust and G. A. Turner, « Real-Time, Continuous Level of Detail Rendering of Height Fields », pp. 109-118, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996
- [16] J. Cohen, A. Varshney, D. Manocha, G. Turk, H. Weber, P. Agarwal, F. Brooks and W. Wright, « Simplification Envelopes », pp. 119-128, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH96, 1996

- [17] H. Pfister, J. Hardenbergh, J. Knittel, H. Lauer and L. Seiler, « The VolumePro Real-Time Ray Casting System », pp. 251-260, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH99, 1999
- [18] T. Pavlidis, « Algorithms for Graphics and Image Processing », Springer 1982
- [19] R. Drebin, L. Carpenter and P. Hanrahan, « Volume Rendering », pp. 60-69, Computer Graphics, SIGGRAPH88, Volume 22, Number 4, August 1988
- [20] M. Do Carmo., « Differential geometry of curves and surfaces », Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1976.
- [21] N. Dyn, D. Levin, and S. Rippa., « Algorithms for the construction of data dependent triangulations », Algorithms for Approximation II, J.C. Mason and M.g.Cox, eds. Chapman and Hall, NY, 10(5):185-192, 1990.
- [22] J. Bloomenthal, K. Ferguson, « Polygonization of Non-Manifold Implicit Surfaces », pp. 309-316, ACM Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH'95, 1995
- [23] J. Popovic, H. Hoppe, « Progressive Simplicial Complexes », pp. 217-224, ACM Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH'97, 1997
- [24] M. A. Sabin, « *Parametric Splines in Tension* », Technical Note VTO/MS/160, British Aircraft Corporation, Weybridge, Surrey, England, July 1970
- [25] P. E. Bézier, « *Essai de Définition Numérique des Courbes et des Surfaces Expérimentales* », PhD dissertation, Université Pierre et Marie Curie, Paris, February 1977
- [26] J. D. Foley, A. van Dam, « *Fundamentals of Interactive Computer Graphics* », Addison-Wesley, Reading, Massachusetts, 1982
- [27] W. M. Newman, R. F. Sproull, « *Principles of Interactive Computer Graphics* », second Edition, McGraw-Hill, New-York
- [28] L. L. Schumaker, « *Spline Functions : Basic Theory* », John Wiley & Sons, New-York, 1981
- [29] R. H. Bartels, J. C. Beaty, B. A. Barsky, « *B-splines* », Mathématiques et CAO, vol. 6, Morgan Kaufmann Publishers, 1987, Hermès Publishing, 1988
- [30] R. H. Bartels, J. C. Beaty, B. A. Barsky, « *Béta-splines* », Mathématiques et CAO, vol. 7, Morgan Kaufmann Publishers, 1987, Hermès Publishing, 1988
- [31] S. A. Coons, « *Surfaces for Computer Aided Design* », Projet MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, 1964
- [32] J. F. Blinn, « *A Generalization of Algebraic Surface Drawing* », ACM Transactions on Graphics, 1(3), July 1982
- [33] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa and K. Omura, « *Object Modeling by Distribution Function and a Method of Image Generation* », Journal of Electronic Communications, 1985
- [34] G. Wyvill, C. McPheeters and B. Wyvill, « Data Structures for Soft Objects », The Visual Computer, 2(4), August 1986
- [35] S. Muraki, « Volumetric Shape Description of Range Data using « Blobby Model » », Computer Graphics, 25(4), pp. 227-235, July 1991
- [36] T.A. Galyean and J.F. Hughes. « *Sculpting : An interactive volumetric modeling technique.* » Computer Graphics, 25(4) :267-274, July 1991. Proceedings of SIGGRAPH'91 (Las Vegas, Nevada, July 1991).

- [37] S.W. Wang and A.E. Kaufman. « *Volume sculpting* ». Computer Graphics, pages 151–156, 1995. Proceedings, Symposium on Interactive 3D graphics.
- [38] E. Ferley, M-P. Cani, J-D. Gascuel, « *Practical Volumetric Sculpting* », Proceedings of the Fourth International Workshop on Implicit Surfaces'99, pp. 51-58, ACM Press, Talence, France, 13-15 September 1999

### 3. TOUR D'HORIZON DE LA RECONSTRUCTION SURFACIQUE.

- [1] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, « *Surface Reconstruction from Unorganized Points* », Computer Graphics, 26 (2), pp. 71-78, July 1992
- [2] S. Pontier, B. Shariat and D. Vandorpe, « *Shape Reconstruction using Skeleton-Based Implicit Surface* », proceedings of 9<sup>th</sup> Canadian Conference on Computational Geometry, pp. 205-210, 1998
- [3] C. T. Lim, G. M. Turkiyyah, M. A. Ganter and D. W. Storti, « *Implicit Reconstruction of Solids from Cloud Point Sets* », Solid Modeling'95, pp. 393-402, ACM, 1995
- [4] F. Lazarus, A. Verroust, « *Décomposition Cylindrique de Polyèdre et Courbe Squelette* », proceedings of 3<sup>èmes</sup> Journées AFIG, pp. 255-264, Novembre 1995
- [5] D. Terzopoulos, « *Dynamic 3D Models with Local and Global Deformations : Deformable Superquadrics* », IEEE Transactions on PAMI, 13(7), pp. 703-714, July 1991
- [6] E. Polak, « *Computational Methods in Optimization : A Unified Approach* », Mathematics in Science and Engineering, vol. 77, Edited by Richard Bellman, Academic Press, 1971
- [7] J-K. Hao, P. Galinier and M. Habib, « *Métaheuristique pour l'Optimisation Combinatoire et l'Affectation sous Contraintes* », Rapport de Synthèse LGI2P EMA-ERIEE/LIRMM URA CNRS, [hao@eerie.fr](mailto:hao@eerie.fr)
- [8] P. Bézier, « *Courbes et Surfaces* », Mathématiques et CAO, vol. 4, Hermès Publishing, 1986
- [9] J. Cartier, G. Laporte, R. Parent and J-M. Picard, « *Optimisation* », Méthodes Quantitatives, vol. 8, Editions Sciences et Cultures Inc., Montreal, 1977
- [10] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, « *Numerical Recipes in C : The Art of Scientific Computing Second Edition* », Cambridge University Press, 1988-1992
- [11] J. Verschelde, « *Homotopy Continuation Methods for Solving Polynomials Systems* », PhD. Thesis, U.D.C. 681.3 F21, Katholieke Universiteit Leuven, May 1996
- [12] P. Pelletier, « *Techniques Numériques appliquées au Calcul Scientifique* », Masson et Cie. Editeurs, 1971
- [13] A. H. Barr, « *Global and Local Deformations of Solid Primitives* », Computer Graphics, 18(3), pp.21-30, July 1984
- [14] W. Metropolis, A. Roenbluth, M. Rosenbluth, A. Teller, E. Teller, « *Equation of the State Calculations by Fast Computing Machines* », Journal of Chemical Physics, 21, pp. 1087-1092, 1953
- [15] S. Lundy, A. Mees, « *Convergence of an Annealing Algorithm* », Mathematical Programming, 34, pp. 111-124, 1986
- [16] G. Ducek, T. Scheuer, « *Threshold Accepting: a General Purpose Optimization Algorithm* », Journal of Computational Physics, 90, pp. 161-175, 1990
- [17] D. Connolly, « *An Improved Annealing Scheme for the QAP* », European Journal of Operational Research, 46, pp. 93-100, 1990
- [18] E. H. L. Aarts, J. K. Lenstra, « *Local Search in Combinatorial Optimization* », John Wiley & Sons, 1997

- [19] C. Koulamas, S. R. Anthony and R. Jean, « *A Survey of Simulated Annealing Application to Operations Research Problem* », Omega, 22, pp. 41-56, 1994
- [20] R. V. Vidal, « *Applied Simulated Annealing* », Lecture Notes in Economics and Mathematical Systems, 396, Springer-Verlag, Berlin, 1993
- [21] F. Glover, « *Future Paths for Integer Programming and Links to Artificial Intelligence* », Computers and Operations Research, 13, pp. 533-549, 1986
- [22] P. Hansen, « *The Steepest Ascent Mildest Descent Heuristic for Combinatorial Programming* », Congress on Numerical Methods in Combinatorial Optimization, Capri, Italie, 1986
- [23] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce and D. J. Kriegman, « *Parameterized Families of Polynomials for Bounded Algebraic Curve and Surface Fitting* », IEEE Transactions on PAMI, 16(3), pp. 287-303, March 1994
- [24] D. R. Forsey, R. H. Bartels, « *Hierarchical B-spline Refinement* », Computer Graphics, 22(3), pp. 205-212, Proceedings of SIGGRAPH88, August 1988
- [25] B. Guo, J. Menon and B. Willette, « *Surface Reconstruction Using Alpha Shapes* », Proceedings of Eurographics'96, 1996
- [26] S. Sclaroff, A. Pentland, « *Generalized Implicit Functions for Computer Graphics* », Computer Graphics, 25(4), pp. 247-250, July 1991
- [27] M. Kass, A. P. Witkin, and D. Terzopoulos, « *Snakes: Active Contour Models* », IJCV, No. 4, pp. 321-331, January 1988
- [28] J. Bloomenthal, « *Surface Tiling* », Implicit Surfaces, Chapter 4
- [29] P. Ning, J. Bloomenthal, « *An Evaluation of Implicit Surface Tilers* »
- [30] P. Borianne, M. Jaeger, « *Polygonisation Réversible en Imagerie Médicale. Application à la Visualisation de Divers Tissus Anatomiques* », Actes CARI'94
- [31] J-D. Boissonnat, « *Shape Reconstruction from Planar Cross Sections* », Computer Vision, Graphics and Image Processing, 44, pp. 1-29, Academic Press, 1988
- [32] J-D. Boissonnat, B. Geiger, « *Three Dimensional Reconstruction of Complex Shapes Based on the Delaunay Triangulation* », RR N°1697, INRIA, Mai 1992
- [33] H. Fuchs, Z. M. Kedem and S. P. Useton, « *Optimal Surface Reconstruction from Planar Contours* », pp. 693-702, Communications of the ACM, Volume 20, Number 10, October 1977
- [34] P. P. Watson, « *Computing the n-dimensional Delaunay Triangulation with Application to Voronoi Polytopes* », pp. 167-172, Computer Journal, Volume 24, Number 2, 1981
- [35] H. Edelsbrunner and R. Seidel, « *Voronoi diagrams and arrangements* », pp. 25-44, Discrete Comput. Geom. 1, 1986.
- [36] G. T. Herman, H. K. Liu, « *Three Dimensional Display of Human Organs from Computed Tomograms* », pp. 1-21, Computer Graphics and Image Processing, Volume 9, Number 1, January 1979
- [37] W. E. Lorensen, H. E. Cline, « *Marching Cubes : A High Resolution 3D Surface Construction Algorithm* », pp. 163-169, Computer Graphics, Volume 21, Number 4, July 1987
- [38] G. M. Treece, R. W. Prager and A. H. Gee, « *Regularised Marching Tetrahedra : Improved Iso-Surface Extraction* », CUED/F-INFENG/TR 333, Cambridge University Engineering Department, September 1998

- [39] P. Shirley, A. Tuckman, « A Polygonal Approximation to Direct Scalar Volume Rendering », pp. 63-70, Computer Graphics, Volume 24, Number 5, November 1990
- [40] H.E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford and B.C. Teeter, « Two Algorithms for the Three Dimensional Reconstruction of Tomograms », pp. 320-327, Med. Phys., 15 (3), Am. Assoc. Phys. Med, May/June 1988
- [41] M. J. Dürst, « Letters : Additional Reference to 'Marching Cubes' », pp. 72-73, ACM Computer Graphics, Volume 22, Number 4, 1988
- [42] K. Shimada, D. C. Gossard, « Bubble Mesh : Automated Triangular Meshing of Non-Manifold Geometry by Sphere Packing », pp. 409-419, ACM Proceedings of Solid Modeling'95, 1995
- [43] W. Seibold, G. Wyll., « Two many triangles. » , p. 242, Computer Graphics, SIGGRAPH'98, Conference abstracts and applications, 1998
- [44] T. Stander, J. C. Hart, « Guaranteeing the Topology of an Implicit Surface Polygonization », Computer Graphics, 1997
- [45] S. Muraki, « *Volumetric Shape Description of Range Data using «Blobby Model»* », Computer Graphics, SIGGRAPH91, 25(4), pp. 227-235, July 1991
- [46] E. Bittar, N. Tsingos and M-P Gascuel, « *Automatic Reconstruction of Unstructured 3D Data: Combining a Medial Axis and Implicit Surfaces.* », Proceedings of Eurographics'95, september 1995
- [47] E. Ferley, M-P. Cani-Gascuel and D. Attali, « *Skeletal Reconstruction of Branching Shapes.* », Computer Graphics Forum, Volume 16 (15), december 1997
- [48] N. Tsingos, E. Bittar and M-P. Gascuel, « *Semi-automatic Reconstruction of Implicit Surfaces for Medical Applications.* », Computer Graphics International'95, Leeds, UK, June 1995
- [49] C. Guiard. « *Modèles de surfaces déformables définis et contraints par un ensemble d'informations structurelles.* », PhD thesis, Université de la Méditerranée, Marseille, 2000.
- [50] H. Delingette, « *General Object Reconstruction based on Simplex Meshes.* », Rapport de recherche N°3111, Thème 3, INRIA, Février 1997



#### **4. TOUR D'HORIZON DE LA RECONNAISSANCE DE FORMES.**

- [1] P. Ferraro, C. Godin, « *Une Méthode de Comparaison Structurale d'Arborescences non Ordonnées* », Document de Travail du Programme Modélisation des Plantes N°1-98, CIRAD, Avril 1998
- [2] C. Berge, « *Graphs* », North Holland, 3<sup>rd</sup> Edition, 1991
- [3] M. Gondran, M. Minoux, « *Graphes et Algorithmes* », Eyrolles, 1990
- [4] S.-Y. Lu, « *A tree-to-tree Distance and its Applications to Cluster Analysis* », IEEE Transactions on PAMI, 1(2), pp. 219-224, 1979
- [5] L. Miclet, « *Méthodes Structurelles pour la Reconnaissance de Formes* », Eyrolles, Paris, 1984
- [6] K. Zhang, « *A Constrained Edit Distance between Unordered Labeled Trees* », Algorithmica, 15, pp. 205-222
- [7] A. G. Mamistvalov, « *n-Dimensional Moment Invariants and Conceptual Mathematical Theory of Recognition n-Dimensional Solids* », IEEE Transactions on PAMI, 20(8), pp. 819-831
- [8] D. L. Borges, R. B. Fisher, « *Class-Based Recognition of 3D Objects Represented by Volumetric Primitives* », British Machine Vision Conference, 1996
- [9] D. L. Borges, « *Recognizing Three-Dimensional Objects Using Parameterized Volumetric Models* », PhD. Thesis, Dept. Artificial Intelligence, University of Edinburgh, 1995
- [10] L. A. Zadeh, « *Fuzzy Sets as a Basis for a Theory of Possibility* », Fuzzy Sets and Systems, 1(1), pp. 3-28, 1978
- [11] I. Biederman, « *Recognition-by-Components : A Theory of Human Image Understanding* », Psychological Review, 94, pp. 115-147, 1987
- [12] M. Näf, O. Kübler, R. Kikinis, M. E. Shenton and G. Székely, « *Characterization and Recognition of 3D Organ Shape in Medical Image Analysis Using Skeletonization* », IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, Hotel Nikko, San Francisco, 1996
- [13] S. Carlsson, « *View Variation and Linear Invariants in 2D and 3D* », Computational Vision and Active Perception Laboratory, Stockholm, Sweden, TRITA – NA – P 9522, CVAP 184
- [14] S. Sclaroff, « *Deformable Prototypes for Encoding Shape Categories in Image Databases* », Pattern Recognition, 30(4), Special Issue on Image Database, April 1997
- [15] D. Forsyth, J. L. Mundy, A. Zisserman, C. Coelho, A. Heller and C. Rothwell, « *Invariant Descriptors for 3D Object Recognition and Pose* », IEEE Transactions on PAMI, 13(10), pp. 971-991, October 1991
- [16] R. Alferez, Y.-F. Wang, « *Geometric and Illumination Invariants for Object Recognition* », IEEE Transactions on PAMI, 21(6), pp. 505-535, June 1999
- [17] E. Cox, « *The Fuzzy Systems Handbook* », Academic Press, 1994
- [18] S. Haykin, « *Neural Networks* », MacMillan, 1994
- [19] L. A. Zadeh, « *Fuzzy Sets* », Information and Control, 8, pp. 338-353, 1965
- [20] S. Sclaroff and A. Pentland, « *Modal Matching for Correspondence and Recognition* », IEEE Transactions on PAMI, June 1995
- [21] D. R. Wilson, T. R. Martinez, « *Improved Heterogeneous Distance Functions* », Journal of Artificial Intelligence Research, 6, pp. 1-34, January 1997
- [22] R. Hecht-Nielsen, « *Neurocomputing* », Reading, MA : Addison-Wesley
- [23] B. D. Ripley, « *Pattern Recognition and Neural Networks* », Cambridge University Press

- [24] P. Schyns, R. L. Goldstone and J.-P. Thibaut, « *The Development of Features in Object Concepts* », Behavioral and Brain Sciences, 21(1), pp. 1-54, 1998
- [25] M. D. Adler, « *An Introduction to Pattern Recognition : Statistical, Neural Net and Syntactic Methods of Getting Robots to See and Hear* », <http://ciips.ee.uwa.edu.au/~mike/PatRec/PatRec.html>, 1997
- [26] T. M. Mitchell, « *The Need for Biases in Learning Generalizations* », Readings in Machine Learning, J. W. Shavlik and T. G. Dietterich Editors, 1980
- [27] C. Schaffer, « *A Conservation Law for Generalization Performance* », Proceedings of the Eleventh International Conference on Machine Learning (ML'94), Morgan Kaufmann, 1994
- [28] P. Turney, « *Exploiting Context when Learning to Classify* », Proceedings of the European Conference on Machine Learning, pp. 402-407, Vienna, Austria, Springer-Verlag, 1993
- [29] D. H. Wolpert, « *On Overfitting Avoidance as Bias* », Technical Report SFI TR 92-03-5001, Santa Fe, The Santa Fe Institute, 1993
- [30] D. Wettscherek, D. W. Aha and T. Mohri, « *A Review and Comparative Evaluation of Feature Weighting Methods for Lazy Learning Algorithms* », Technical Report AIC-95-012, Washington, DC, Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence, 1995
- [31] C. Stanfill, D. Waltz, « *Toward Memory-Based Reasoning* », Communications of the ACM, 29, pp. 1213-1228, December 1986
- [32] D. Ventura, « *On Discretization as a Preprocessing Step for Supervised Learning Models* », Master's Thesis, Department of Computer Science, Brigham Young University
- [33] D. Ventura, T. R. Martinez, « *An Empirical Comparison of Discretization Methods* », Proceedings of the Tenth International Symposium on Computer and Information Sciences, pp. 443-450, 1995
- [34] A. Nigrin, « *Neural Networks for Pattern Recognition* », p. 11, Cambridge, MA : The MIT Press, 1993
- [35] P.C. Mahalanobis. On the Generalized Distance in Statistics. *National Institute of Science in India*, 12:49-55, 1936.
- [36] P. Devijver, J. Kittler, « *Pattern Recognition. A statistical Approach.* », Prentice Hall 82.
- [37] G. Gaillat, « *Méthodes statistiques de Reconnaissance des Formes.* », ENSTA 83.
- [38] L. Miclet, « *Méthodes structurelles pour la Reconnaissance des Formes.* », Eyrolles 84.



## 5. LE E-SQUELETTE.

- [1] Frédéric Banégas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Partitionnement Hiérarchique d'un Nuage de Points par Squelette Ellipsoïdal* », 1998, 6ièmes Journées de l'Association Française d'Informatique Graphique, Dunkerque, France.
- [2] Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Application of Ellipsoidal Skeleton to Organ Geometric Modeling from CT-scan Images.* », 1999, SPIE Conference on Medical Imaging '99, San Diego, California.
- [3] Frédéric Banégas, Dr. François Canovas, Dominique Michelucci, Marc Roelens, Marc Jaeger, « *Hierarchical Pattern Analysis and Recognition Using Ellipsoidal Skeleton.* », 2000, Fourth International Conference in Computer Graphics and Artificial Intelligence, 3IA'2000, Limoges, France.
- [4] Murray R. Spiegel, « *Théorie et Applications de la Mécanique Générale.* », Série Schaum, Mac Graw Hill, 1972.
- [5] D. Forsyth, J. L. Mundy, A. Zisserman, C. Coelho, A. Heller and C. Rothwell, « *Invariant Descriptors for 3D Object Recognition and Pose.* », IEEE Transactions on PAMI, 13 :10, pp. 971-991, October, 1991.
- [6] Y. Guédon, « *Variables Aléatoires Discrètes, Processus Stochastiques à Temps Discret et à Espace d'Etats Discret.* », Rapport Interne, CIRAD, Programme Modélisation des Plantes, 1999.
- [7] A. Leonardis, A. Jaklic and F. Solina, « *Superquadrics for Segmenting and Modeling Range Data.* », IEEE Transactions on PAMI, 19 :11, pp. 1289-1295, November, 1997.
- [8] P. Bajcsy, N. Ahuja, « *Location- and Density-Based Hierarchical Clustering Using Similarity Analysis.* », IEEE Transactions on PAMI, 20 :9, pp. 1011-1015, Short Papers, September, 1998.
- [9] I. Pitas, A. N. Venetsanopoulos, « *Morphological Shape Decomposition.* », IEEE Transactions on PAMI, 12 :1, pp. 38-45, January, 1990.
- [10] P. Borianne, « *Cepha 3Dt-version 2.0.* », Chapitre « Outils Mathématiques. », CIRAD-MEB, 1998.
- [11] M. Murphy, S. Skiena, « *A Study of Data Structures for Orthogonal Range and Nearest Neighbor Queries in High Dimensional Spaces.* », CSE 523/524 Master's Project, Department of Computer Science, State University of New York at Stony Brook, NY 11790-4400.
- [12] « *Recherche Opérationnelle/Operations Research.* », 0399-0559/98/04, Elsevier, Paris, 1998.
- [13] E. Diday, « *Optimisation en Classification Automatique et Reconnaissance des Formes.* », RAIRO, Novembre, 1972.
- [14] M. Jambu, « *Classification Automatique pour l'Analyse des Données.* », Dunod, 1978.
- [15] S. Regnier, « *Sur Quelques Aspects Mathématiques des Problèmes de Classification.* », ICC Bull., vol. 4, pp. 175-191, 1965.
- [16] K. S. Younis, « *Weighted Mahalanobis Distance for Hyper-Ellipsoidal Clustering.* », PhD. Thesis, Department of the Air Force Institute of Technology, RJAF, AFIT/GE/ENG/96D-22, Wright-Patterson Air Force, December, 1996.

- [17] D. R. Wilson, T. R. Martinez, « *Improved Heterogeneous Distance Functions.* », Journal of Artificial Intelligence Research, vol. 6, pp. 1-34, AI Access Foundation and Morgan Kaufmann Publishers, 1997.
- [18] B. D. Ripley, « *Neural Networks and Related Methods for Classification.* », Journal of The Royal Statistical Society, Series B, vol. 56, pp. 409-456, 1994.
- [19] S. Byers, A. E. Raftery, « *Nearest Neighbor Clutter Removal for Estimating Features in Spatial Point Processes.* », Journal of the American Statistical Association, 1998.
- [20] R. E. Kass, A. E. Raftery, « *Bayes Factor.* », Journal of the American Statistical Association, vol. 90, pp. 773-795, 1995.
- [21] A. J. Scott, M. J. Symons, « *Clustering Methods based on Likelihood Ratio Criteria.* », Biometrics, vol. 27, pp. 387-397, 1971.
- [22] L. Kaufman, P. J. Rousseuw, « *Finding Groups in Data.* », Wiley, 1990.
- [23] J. A. Hartigan, « *Clustering Algorithms.* », Wiley, 1975.
- [24] P. Kamina, J-P. Francke, « *Anatomie : Introduction à la Clinique. Arthrologie des Membres.* », Maloine S.A. Editeurs, 1988.
- [25] J-P. Chevrel, J-P. Guéraud and J-B. Lévy, « *Anatomie Générale.* », 4<sup>ème</sup> Edition, Masson, 1986.
- [26] N. Ueda, « *Optimal Linear Combination of Neural Networks for Improving Classification Performance.* », IEEE Transactions on PAMI, 22 :2, pp. 207-215, February, 2000.

## 6. RECONSTRUCTION GÉOMÉTRIQUE PAR E-SQUELETTE.

- [1] E. Castillo and A. Iglesias, « *Some Characterizations of Families of Surfaces Using Functional Equations.* », ACM Transactions on Graphics, vol. 16, No. 3, pp. 296-318, July 1997
- [2] J. F. Blinn, « *A Generalization of Algebraic Surface Drawing.* », ACM Transactions on Graphics, vol. 1, No. 3, pp. 235-256, July 1982
- [3] F. Banégas, D. Michelucci, M. Roelens and M. Jaeger, « *Automatic Adaptive Surface Reconstruction from Ellipsoidal Skeleton.* », 1999, The Fourth International Workshop on Implicit Surfaces '99, Bordeaux, France.
- [4] F. Banégas, D. Michelucci, M. Roelens and M. Jaeger, « *Automatic Extraction of Significant Features from 3D Point Clouds by Ellipsoidal Skeleton. Applications in Vision and Geometrical Characterization.* », 1999, International Conference in Visual Computing '99, Goa, India.
- [5] B. Wyvill, G. Wyvill, « *Field Functions for Implicit Surfaces.* », The Visual Computer, 5, pp. 75-82, Springer-Verlag, 1989
- [6] D. Rossin, M-P. Cani-Gascuel, « *Plateforme d'Expérimentation pour la Modélisation par Surfaces Implicites.* », Journées « Modeleurs Géométriques », Grenoble, 17-19 septembre 1997
- [7] A. Pasko, V. Adzhiev, A. Sourin and V. Savchenko, « *Function Representation in Geometric Modeling : Concepts, Implementation and Applications.* », The Visual Computer, 11, pp. 429-446, Springer-Verlag, 1995
- [8] J. Bloomenthal, K. Shoemake, « *Convolution Surfaces.* », SIGGRAPH'91, Computer Graphics, vol. 25, No. 4, pp. 251-256, July 1991
- [9] T. Duff, « *Interval Arithmetic and Recursive Subdivision for Implicit Functions and Constructive Solid Geometry.* », SIGGRAPH'92, Computer Graphics, vol. 26, No. 2, pp. 131-138, July 1992
- [10] M. Kisters, « *An Extension of the Potential Method to Higher-Order Blendings.* », SIGGRAPH'91, Computer Graphics, vol. 25, No. 4, pp. 329-337, July 1991
- [11] H. Hoppe, « *Progressive Meshes.* », SIGGRAPH'96, pp. 99-108, 1996.
- [12] R. Fletcher, « *Conjugate Gradient Methods for Indefinite Systems.* », Numerical Analysis Dundee, Springer-Verlag, 1995
- [13] A. Barr, « *Global and Local Deformations of Solid Primitives.* », Computer Graphics, vol. 18, pp. 21-30, ACM Press, 1984
- [14] V. Burdin, C. Roux, C. Lefèvre, E. Stindel, « *Modeling and Analysis of 3D Elongated Shapes with Applications to Long Bone Morphometry.* », IEEE Transactions on Medical Imaging, vol. 15, pp. 79-91, IEEE, February 1996
- [15] G. Danuser, M. Stricker, « *Parametric Model Fitting : From Inlier Characterization to Outlier Detection.* », IEEE Transactions on PAMI, vol. 20, pp. 263-280, IEEE, March 1998
- [16] W. E. Lorensen, H. E. Cline, « *Marching Cubes : A High Resolution 3D Surface Algorithm.* », Computer Graphics, vol. 21, pp. 163-169, ACM Press, July 1987
- [17] A. Leonardis, A. Jaklic and F. Solina, « *Superquadrics for Segmenting and Modeling Range Data.* », IEEE PAMI, vol. 19, pp. 1289-1295, IEEE, November 1997

- [18] C. T. Lim, G. M. Turkiyyah, M. A. Ganter and D. W. Stortri, « *Implicit Reconstruction of Solids from Cloud Point Sets.* », Solid Modelling'95, pp. 393-402, ACM Press, 1995
- [19] S. Muraki, « *Volumetric Shape Description of Range Data using Blobby Model.* », Computer Graphics, vol. 25, pp. 227-235, ACM Press, 1991
- [20] M. L. Rhodes, « *Computer Graphics in Medicine : A Complex Partnership.* », IEEE Computer Graphics and Applications, pp. 22-28, IEEE, January-February 1997
- [21] S. Sclaroff, A. Pentland, « *Generalized Implicit Functions for Computer Graphics.* », vol. 25, pp. 247-250, ACM Press, July 1991
- [22] V. V. Savchenko, A. A. Pasko, O. G. Okunev and T. L. Kunii, « *Function Representation of Solids Reconstructed from Scattered Surface Points and Contours.* », Eurographics'95, Blackwell Publishers, 1995
- [23] R. Durikovic, K. Kaneda and H. Yamashita, « *Reconstructing a 3D Structure with Multiple Deformable Solid Primitives.* », Computer and Graphics, vol. 21, pp. 611-622, Pergamon, 1997
- [24] J. Bloomenthal, « *An Implicit Surface Polygonizer.* », Graphics Gems IV, Paul Heckbert, Academic Press, New York, 1994
- [25] M. Hall, J. Warr, « *Adaptive Polygonization of Implicitly Defined Surfaces.* », IEEE Computer Graphics and Applications, IEEE Computer Society Press, vol. 10, pp. 33-42, November 1990
- [26] D. Metaxas, D. Terzopoulos, « *Dynamic Deformation of Solid Primitives with Constraints.* », Proceedings of SIGGRAPH'92, vol. 26, ACM SIGGRAPH, pp. 309-312, July 1992
- [27] G. Nielson, B. Hamann, « *The Asymptotic Decider : Resolving the Ambiguity in Marching Cubes.* », Proceedings of Visualization'91, Nielson G. M., Rosenblum L. Eds., pp. 83-91, october 1991
- [28] G. D. Purvis, C. Culberson, « *On the Graphical Display of Molecular Electrostatic Force-Fields and Gradients of the Electron Density.* », NATO Conference, pp. 317-332, July 1985
- [29] V. Pratt, « *Direct Least-Squares Fitting of Algebraic Surfaces.* », Proceedings of SIGGRAPH'87, vol. 21, ACM SIGGRAPH, pp. 145-152, 1987
- [30] A. Ricci, « *A Constructive Geometry for Computer Graphics.* », The Computer Journal, vol. 16, pp. 157-160, May 1973
- [31] G. Taubin, « *Distance Approximation for Rasterizing Implicit Curves.* », ACM Transactions on Graphics, vol. 13, pp. 3-42, 1994
- [32] B. Wyvill, K. van Overveld, « *Polygonization of Implicit Surfaces with Constructive Solid Geometry.* », Journal of Shape Modeling, vol. 2, World Scientific Publishing, 1997
- [33] G. Wyvill, « *Data Structures for Soft Objects.* », The Visual Computer, vol. 2, pp. 227-234, August 1986
- [34] J-K. Hao, P. Galinier and M. Habib, « *Métaheuristique pour l'Optimisation Combinatoire et l'Affectation sous Contraintes.* », Research Report, EMA-EERIE
- [35] J-K. Hao, R. Dorne, « *Empirical Studies of Heuristic Local Search for Constraint Solving.* », Proceedings of CP-96, LNCS 1118, pp. 194-208, Cambridge, MA, USA, 1996
- [36] F. Glover, « *Tabu Search ; Part I.* », ORSA Journal on Computing, vol. 1, No. 3, pp. 190-206, 1989

- [37] F. Glover, « *Tabu Search ; Part II.* », ORSA Journal on Computing, vol. 2, No. 1, pp. 4-32, 1989
- [38] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, « *Numerical Recipes in C : The Art of Scientific Computing Second Edition.* », Cambridge University Press, 1988-1992
- [39] S. Chemouny, H. Joyeux, B. Masson, F. Borne, M. Jaeger and O. Monga, « *Advanced 3D image processing techniques for liver and hepatic tumors location and volumetry.* », Medical Imaging 1999 - Image Processing, K.M. Hanson (Ed.), San Diego, USA, SPIE The International Society for Optical Engineering, SPIE Proceedings Series (3661-1), pp. 761-771, 20-26/02/1999
- [40] Z. Ka, I. Alesi, « *Blending Techniques for Procedural Implicit Surfaces.* », master's thesis, Department of Computer Science, the University of Calgary, Alberta, 1991
- [41] J. Dauzat, F. Sillion, C. Nougier and C. Damez, « *Specific methods for radiative transfer simulations on multi-scal plant mock-ups.* », Second International workshop on Functional-Structural Tree Models, Clermont-Ferrand, France, 1998
- [42] Hitoshi Nishimura, Makoto Hirai, Toshiyuki Kawai, Toru Kawata, Isao Shirakawa and Koichi Omura, « *Object Modeling by Distribution Function and a Method of Image Generation* », Transactions of the Institute of Electronics and Communication Engineers of Japan, 1985, vol. J68-D, no. 4, pp. 718-725, in Japanese (English translation by Takao Fujuwara, Advanced Studies in Computer Aided Art and Design, Middlesex Polytechnic, England, 1989)
- [43] M. Desbrun, N. Tsingos and M-P. Gascuel, « *Adaptive Sampling of Implicit Surfaces for Interactive Modeling and Animation.* », Computer Graphics Forum, December 1996

## 7. RECONNAISSANCE DE FORMES PAR E-SQUELETTE.

- [1] F. Banégas, Dr. F. Canovas, D. Michelucci, M. Roelens et M. Jaeger, « *Hierarchical Pattern Analysis and Recognition Using Ellipsoidal Skeleton.* », 2000, Fourth International Conference in Computer Graphics and Artificial Intelligence, 3IA'2000, Limoges, France.
- [2] F. Canovas, F. Banégas, C. Cyteval, M. Jaeger, A. Dimeglio, C. Sultan, F. Bonnel, « *Carpal Bone Maturation Assessment by Image Analysis from CT-scans.* », International Journal of Experimental and Clinical Endocrinology, Hormone Research, Karger Eds, 2000.
- [3] D. de Werra, « *Elements de Programmation Linéaire avec Application aux Graphes.* », Presses Polytechniques Romandes, 1990.
- [4] A. K. Jain, R. P. W. Duin and J. Mao, « *Statistical Pattern Recognition : A Review.* », IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), pp. 4-34, January, 2000.
- [5] T. M. Cover, J. M. Van Campenhout, « *On the Possible Orderings in the Measurement Selection Problem.* », IEEE Transactions on Systems, Man and Cybernetics, 7(9), pp. 657-661, September, 1977.
- [6] A. K. Jain, D. Zongker, « *Feature Selection : Evaluation, Application and Small Sample Performance.* », IEEE Transactions on PAMI, 19(2), pp. 153-158, February, 1997.
- [7] D. Shen, H. S. Horace, K. T. Cheung and E. A. Teoh, « *Symmetry Detection by Generalized Complex (GC) Moments : A Close-Form Solution.* », IEEE Transactions on PAMI, 21(5), May, 1999.
- [8] H. Hügli, C. Schütz, « *Geometric Matching of 3D Objects : Assessing the Range of Successful Configurations.* », International Conference of Recent Advances in 3D Digital Imaging and Modeling, May 12-15, Ottawa, Ontario, Canada, 1997.
- [9] K. Zhang, « *A Constrained Edit Distance Between Unordered Labeled Trees.* », Algorithmica, 15, pp. 205-222, Springer-Verlag New York Inc., 1996.
- [10] J. Martin, A. Pentland, S. Sclaroff and Ron Kikinis, « *Characterization of Neuropathological Shape Deformations.* », IEEE Transactions on PAMI, 20(2), pp. 97-112, February, 1998.
- [11] E. P. Simoncelli, « *A Rotation-Invariant Pattern Signature.* », Proceedings of 3<sup>rd</sup> IEEE International Conference on Image Processing, III, pp. 185-188, IEEE Signal Processing Society, Lausanne, Switzerland, 16-19 september, 1996.
- [12] L. R. Ford Jr., D. R. Fulkerson, « *Maximal Flow through a Network.* », Canadian Journal of Mathematics, 8, pp. 399-404, 1956.
- [13] L. R. Ford Jr., D. R. Fulkerson, « *Flow in Networks.* », Princeton University Press, Princeton, NJ, 1962.
- [14] J. Edmonds, R. M. Karp, « *Theoretical improvements in algorithmic efficiency for network flow problems.* », Journal of the ACM, 19(2):248-264, April 1972.
- [15] Z. Galil, « *Efficient Algorithms for Finding Maximum Matching in Graphs.* », Computing Surveys, vol. 18, No. 1, March 1986.
- [16] Z. Galil, S. Micali and H. Gabow, « *An  $O(EV \log V)$  Algorithm for Finding a Maximal Weighted Matching in General Graphs.* », SIAM J. COMPUT., vol. 15, No. 1, February 1986.

- [17] D. de Werra, « *Eléments de Programmation Linéaire avec Application aux Graphes.* », Presses Polytechniques Romandes, p. 200, 1990.
- [18] R. Tam, A. Fournier, « *Image Interpolation Using Union of Spheres.* », The Visual Computer, 14, 8/9, pp. 401-414, 1998.
- [19] F. Canovas, « *Etude de la Maturation des Os du Carpe par Analyse d'Image à partir de Coupes Tomodensitométriques du Poignet.* », Mémoire de thèse, Université Montpellier I, U.F.R. de Médecine, Formation doctorale : Endocrinologie cellulaire et moléculaire, Discipline : Anatomie, 24 octobre 2000.









